Module sui_system::validator

Event emitted when a new stake request is received.

Event emitted when a new unstake request is received.

Event emitted when a staked SUI is converted to a fungible staked SUI.

Event emitted when a fungible staked SUI is redeemed.

Invalid proof_of_possession field in ValidatorMetadata

Invalid pubkey_bytes field in ValidatorMetadata

Invalid network_pubkey_bytes field in ValidatorMetadata

Invalid worker_pubkey_bytes field in ValidatorMetadata

Invalid net_address field in ValidatorMetadata

Invalid p2p_address field in ValidatorMetadata

Invalid primary_address field in ValidatorMetadata

Invalid worker_address field in ValidatorMetadata

Commission rate set by the validator is higher than the threshold

Validator Metadata is too long

Intended validator is not a candidate one.

Stake amount is invalid or wrong.

Function called during non-genesis times.

New Capability is not created by the validator itself

Capability code is not valid

Validator trying to set gas price higher than threshold.

Max gas price a validator can set is 100K MIST.

Deactivate this validator's staking pool

Process pending stake and pending withdraws, and update the gas price.

Request to add stake to the validator's staking pool, processed at the end of the epoch.

Request to add stake to the validator's staking pool at genesis

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

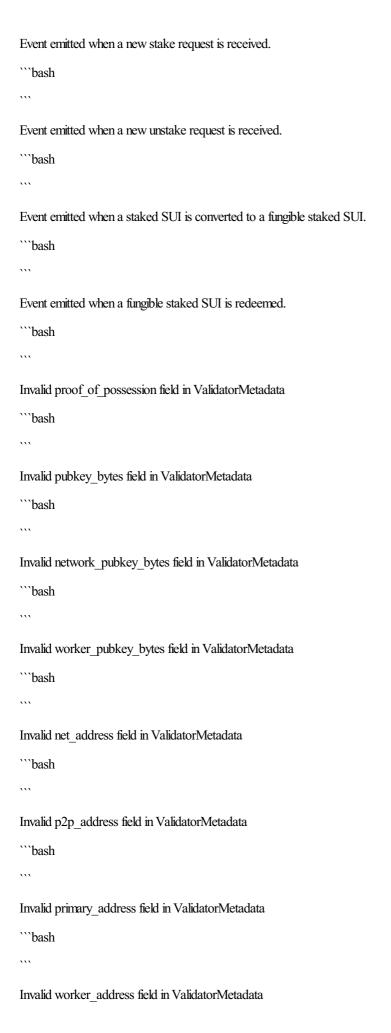Set new gas price for the candidate validator.

Request to set new commission rate for the next epoch.

Set new commission rate for the candidate validator.

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

Process pending stakes and withdraws, called at the end of the epoch.

Returns true if the validator is preactive.

Return the total amount staked with this validator

Return the voting power of this validator.

Set the voting power of this validator, called only from validator_set.

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

Update name of the validator.

Update description of the validator.

Update image url of the validator.

Update project url of the validator.

Update network address of this validator, taking effects from next epoch

Update network address of this candidate validator

Update p2p address of this validator, taking effects from next epoch

Update p2p address of this candidate validator

Update primary address of this validator, taking effects from next epoch

Update primary address of this candidate validator

Update worker address of this validator, taking effects from next epoch

Update worker address of this candidate validator

Update protocol public key of this validator, taking effects from next epoch

Update protocol public key of this candidate validator

Update network public key of this validator, taking effects from next epoch

Update network public key of this candidate validator

Update Narwhal worker public key of this validator, taking effects from next epoch

Update Narwhal worker public key of this candidate validator

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

Helper macro which extracts the value from Some and applies $f to it.

Aborts if validator metadata is valid

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

Struct

```bash
```

```bash
```

Event emitted when a new stake request is received.

```bash
```

Event emitted when a new unstake request is received.

```bash
```

Event emitted when a staked SUI is converted to a fungible staked SUI.

```bash
```

Event emitted when a fungible staked SUI is redeemed.

```bash
```

Invalid proof_of_possession field in ValidatorMetadata

```bash
```

Invalid pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid network_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid worker_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid net_address field in ValidatorMetadata

```bash
```

Invalid p2p_address field in ValidatorMetadata

```bash
```

Invalid primary_address field in ValidatorMetadata

```bash
```

Invalid worker_address field in ValidatorMetadata

```bash
```

Commission rate set by the validator is higher than the threshold

```bash
```

Validator Metadata is too long

```bash
```

Intended validator is not a candidate one.

```bash
```

Stake amount is invalid or wrong.

```bash
```

Function called during non-genesis times.

```bash
```

New Capability is not created by the validator itself

```bash
```

Capability code is not valid

```bash
```

Validator trying to set gas price higher than threshold.

```bash
```

```bash
```

```bash
```

Max gas price a validator can set is 100K MIST.

```bash
```

```
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Deactivate this validator's staking pool

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Process pending stake and pending withdraws, and update the gas price.

```bash
bash (package)
```

```bash
bash (package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Request to add stake to the validator's staking pool at genesis

```bash
bash (package)
```

```bash
bash (package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
bash (package)
```

```bash
bash (package)
```

Set new gas price for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new commission rate for the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Set new commission rate for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash

```

```bash

```

Struct

```bash
```

Event emitted when a new stake request is received.

```bash
```

Event emitted when a new unstake request is received.

```bash
```

Event emitted when a staked SUI is converted to a fungible staked SUI.

```bash
```

Event emitted when a fungible staked SUI is redeemed.

```bash
```

Invalid proof_of_possession field in ValidatorMetadata

```bash
```

Invalid pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid network_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid worker_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid net_address field in ValidatorMetadata

```bash
```

Invalid p2p_address field in ValidatorMetadata

```bash
```

Invalid primary_address field in ValidatorMetadata

```bash
```

```
```

Invalid worker_address field in ValidatorMetadata

```bash
```

Commission rate set by the validator is higher than the threshold

```bash
```

Validator Metadata is too long

```bash
```

Intended validator is not a candidate one.

```bash
```

Stake amount is invalid or wrong.

```bash
```

Function called during non-genesis times.

```bash
```

New Capability is not created by the validator itself

```bash
```

Capability code is not valid

```bash
```

Validator trying to set gas price higher than threshold.

```bash
```

```bash
```

```bash
```

Max gas price a validator can set is 100K MIST.

```bash
```

```
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Deactivate this validator's staking pool

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Process pending stake and pending withdraws, and update the gas price.

```
bash (package)
```

```
bash (package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Request to add stake to the validator's staking pool at genesis

```
bash (package)
```

```
bash (package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```
bash (package)
```

```
bash (package)
```

Set new gas price for the candidate validator.

```
bash (package)
```

```
bash (package)
```

Request to set new commission rate for the next epoch.

```
bash (package)
```

```
bash (package)
```

Set new commission rate for the candidate validator.

```
bash (package)
```

```bash
(package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network public key of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update Narwhal worker public key of this candidate validator

```
bash (package)
```

```
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```
bash (package)
```

```
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash

```

```bash

```

Struct

Event emitted when a new stake request is received.

```bash
```

Event emitted when a new unstake request is received.

```bash
```

Event emitted when a staked SUI is converted to a fungible staked SUI.

```bash
```

Event emitted when a fungible staked SUI is redeemed.

```bash
```

Invalid proof_of_possession field in ValidatorMetadata

```bash
```

Invalid pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid network_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid worker_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid net_address field in ValidatorMetadata

```bash
```

Invalid p2p_address field in ValidatorMetadata

```bash
```

Invalid primary_address field in ValidatorMetadata

```bash
```

Invalid worker_address field in ValidatorMetadata

```bash
```

Commission rate set by the validator is higher than the threshold

```bash
```

Validator Metadata is too long

```bash
```

Intended validator is not a candidate one.

```bash
```

Stake amount is invalid or wrong.

```bash
```

Function called during non-genesis times.

```bash
```

New Capability is not created by the validator itself

```bash
```

Capability code is not valid

```bash
```

Validator trying to set gas price higher than threshold.

```bash
```

```bash
```

```bash
```

Max gas price a validator can set is 100K MIST.

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Deactivate this validator's staking pool

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Process pending stake and pending withdraws, and update the gas price.

```bash
bash (package)
```

```bash
bash (package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Request to add stake to the validator's staking pool at genesis

```bash
bash (package)
```

```bash
bash (package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
bash (package)
```

```bash
bash (package)
```

Set new gas price for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new commission rate for the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Set new commission rate for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Struct

Event emitted when a new unstake request is received.

```bash
```

Event emitted when a staked SUI is converted to a fungible staked SUI.

```bash
```

Event emitted when a fungible staked SUI is redeemed.

```bash
```

Invalid proof_of_possession field in ValidatorMetadata

```bash
```

Invalid pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid network_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid worker_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid net_address field in ValidatorMetadata

```bash
```

Invalid p2p_address field in ValidatorMetadata

```bash
```

Invalid primary_address field in ValidatorMetadata

```bash
```

Invalid worker_address field in ValidatorMetadata

```bash
```

Commission rate set by the validator is higher than the threshold

```bash
```

Validator Metadata is too long

```bash
```

Intended validator is not a candidate one.

```bash
```

Stake amount is invalid or wrong.

```bash
```

Function called during non-genesis times.

```bash
```

New Capability is not created by the validator itself

```bash
```

Capability code is not valid

```bash
```

Validator trying to set gas price higher than threshold.

```bash
```

```bash
```

```bash
```

Max gas price a validator can set is 100K MIST.

```bash
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```bash
bash (package)
```

Deactivate this validator's staking pool

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Process pending stake and pending withdraws, and update the gas price.

```bash
bash (package)
```

```bash
bash (package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Request to add stake to the validator's staking pool at genesis

```bash
bash (package)
```

```bash
bash (package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
bash (package)
```

```bash
bash (package)
```

Set new gas price for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new commission rate for the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Set new commission rate for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

bash (package)

bash (package)

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Struct

Event emitted when a staked SUI is converted to a fungible staked SUI.

```bash
```

```
```

Event emitted when a fungible staked SUI is redeemed.

```bash
```

Invalid proof_of_possession field in ValidatorMetadata

```bash
```

Invalid pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid network_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid worker_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid net_address field in ValidatorMetadata

```bash
```

Invalid p2p_address field in ValidatorMetadata

```bash
```

Invalid primary_address field in ValidatorMetadata

```bash
```

Invalid worker_address field in ValidatorMetadata

```bash
```

Commission rate set by the validator is higher than the threshold

```bash
```

Validator Metadata is too long

```bash
```

Intended validator is not a candidate one.

```bash
```

Stake amount is invalid or wrong.

```bash
```

Function called during non-genesis times.

```bash
```

New Capability is not created by the validator itself

```bash
```

Capability code is not valid

```bash
```

Validator trying to set gas price higher than threshold.

```bash
```

```bash
```

```bash
```

Max gas price a validator can set is 100K MIST.

```bash
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Deactivate this validator's staking pool

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Process pending stake and pending withdraws, and update the gas price.

```bash
bash (package)
```

```bash
bash (package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Request to add stake to the validator's staking pool at genesis

```bash
bash (package)
```

```bash
bash (package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
bash (package)
```

```bash
bash (package)
```

Set new gas price for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new commission rate for the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Set new commission rate for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update p2p address of this candidate validator

```bash (package)
```

```bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update primary address of this candidate validator

```bash (package)
```

```bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update worker address of this candidate validator

```bash (package)
```

```bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update protocol public key of this candidate validator

```bash (package)
```

```bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network public key of this candidate validator

```bash (package)
```

```bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash (package)
```

```bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when

advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Struct

Event emitted when a fungible staked SUI is redeemed.

```bash
```

Invalid proof_of_possession field in ValidatorMetadata

```bash
```

Invalid pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid network_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid worker_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid net_address field in ValidatorMetadata

```bash
```

Invalid p2p_address field in ValidatorMetadata

```bash
```

Invalid primary_address field in ValidatorMetadata

```bash
```

Invalid worker_address field in ValidatorMetadata

```bash
```

Commission rate set by the validator is higher than the threshold

```bash
```

Validator Metadata is too long

```bash
```

Intended validator is not a candidate one.

```bash
```

Stake amount is invalid or wrong.

```bash
```

Function called during non-genesis times.

```bash
```

New Capability is not created by the validator itself

```bash
```

Capability code is not valid

```bash
```

Validator trying to set gas price higher than threshold.

```bash
```

```bash
```

```bash
```

Max gas price a validator can set is 100K MIST.

```bash
```

```
bash (package)
```
```
bash (package)
```
```
bash (package)
```
```
bash (package)
```

Deactivate this validator's staking pool

```
bash (package)
```
```
bash (package)
```
```
bash (package)
```
```
bash (package)
```

Process pending stake and pending withdraws, and update the gas price.

```
bash (package)
```
```
bash (package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```
bash (package)
```
```
bash (package)
```
```
bash (package)
```
```
bash (package)
```
```
bash (package)
```
```
bash (package)
```

Request to add stake to the validator's staking pool at genesis

```
bash (package)
```

```bash
(package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
(package)
```

```bash
(package)
```

Set new gas price for the candidate validator.

```bash
(package)
```

```bash
(package)
```

Request to set new commission rate for the next epoch.

```bash
(package)
```

```bash
(package)
```

Set new commission rate for the candidate validator.

```bash
(package)
```

```bash
(package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Return the total amount staked with this validator

```bash

```

```bash

```

Return the voting power of this validator.

```bash

```
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

bash (package)

bash (package)

Update name of the validator.

bash (package)

bash (package)

Update description of the validator.

bash (package)

bash (package)

Update image url of the validator.

bash (package)

bash (package)

Update project url of the validator.

bash (package)

bash (package)

Update network address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update network address of this candidate validator

bash (package)

bash (package)

Update p2p address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update p2p address of this candidate validator

bash (package)

bash (package)

Update primary address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update primary address of this candidate validator

bash (package)

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```
bash (package)
```
```
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Constants

Invalid proof_of_possession field in ValidatorMetadata

```bash
```

Invalid pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid network_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid worker_pubkey_bytes field in ValidatorMetadata

```bash
```

Invalid net_address field in ValidatorMetadata

```bash
```

Invalid p2p_address field in ValidatorMetadata

```bash
```

Invalid primary_address field in ValidatorMetadata

```bash
```

Invalid worker_address field in ValidatorMetadata

```bash
```

Commission rate set by the validator is higher than the threshold

```bash
```

Validator Metadata is too long

```bash
```

Intended validator is not a candidate one.

```bash
```

Stake amount is invalid or wrong.

```bash
```

Function called during non-genesis times.

```bash
```

New Capability is not created by the validator itself

```bash
```

Capability code is not valid

```bash
```

Validator trying to set gas price higher than threshold.

```bash
```

```bash
```

```bash
```

Max gas price a validator can set is 100K MIST.

```bash

```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Deactivate this validator's staking pool

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Process pending stake and pending withdraws, and update the gas price.

```
bash (package)
```

```
bash (package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Request to add stake to the validator's staking pool at genesis

```
bash (package)
```

```
bash (package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```
bash (package)
```

```
bash (package)
```

Set new gas price for the candidate validator.

```
bash (package)
```

```
bash (package)
```

Request to set new commission rate for the next epoch.

```
bash (package)
```

```
bash (package)
```

Set new commission rate for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

bash (package)

bash (package)

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```bash (package)
```

```bash (package)
```

Update project url of the validator.

```bash (package)
```

```bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network address of this candidate validator

```bash (package)
```

```bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update p2p address of this candidate validator

```bash (package)
```

```bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update primary address of this candidate validator

```bash (package)
```

```bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update worker address of this candidate validator

```bash (package)
```

```bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update protocol public key of this candidate validator

```bash (package)
```

```bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Deactivate this validator's staking pool

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Process pending stake and pending withdraws, and update the gas price.

```bash
(package)
```

```bash
(package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Request to add stake to the validator's staking pool at genesis

```bash
(package)
```

```bash
(package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
(package)
```

```bash
(package)
```

Set new gas price for the candidate validator.

```bash
(package)
```

```bash
(package)
```

Request to set new commission rate for the next epoch.

```bash
(package)
```

```bash
(package)
```

Set new commission rate for the candidate validator.

```bash
(package)
```

```bash
 (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
 (package)
```

```bash
 (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
 (package)
```

```bash
 (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```bash (package)```

```bash (package)```

Update name of the validator.

```bash (package)```

```bash (package)```

Update description of the validator.

```bash (package)```

```bash (package)```

Update image url of the validator.

```bash (package)```

```bash (package)```

Update project url of the validator.

```bash (package)```

```bash (package)```

Update network address of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update network address of this candidate validator

```bash (package)```

```bash (package)```

Update p2p address of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update p2p address of this candidate validator

```bash (package)```

```bash (package)```

Update primary address of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update primary address of this candidate validator

```bash (package)```

```bash (package)```

Update worker address of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update worker address of this candidate validator

```bash (package)```

```bash (package)```

Update protocol public key of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update protocol public key of this candidate validator

```bash (package)```

```bash (package)```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

```bash
(package)
```

```bash
(package)
```

Deactivate this validator's staking pool

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Process pending stake and pending withdraws, and update the gas price.

```bash
(package)
```

```bash
(package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Request to add stake to the validator's staking pool at genesis

```bash
(package)
```

```bash
(package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
(package)
```

```bash
(package)
```

Set new gas price for the candidate validator.

```bash
(package)
```

```bash
(package)
```

Request to set new commission rate for the next epoch.

```bash
(package)
```

```bash
(package)
```

Set new commission rate for the candidate validator.

```bash
(package)
```

```bash
(package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

bash (package)

bash (package)

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```bash (package)
```

Update project url of the validator.

```bash (package)
```

```bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network address of this candidate validator

```bash (package)
```

```bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update p2p address of this candidate validator

```bash (package)
```

```bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update primary address of this candidate validator

```bash (package)
```

```bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update worker address of this candidate validator

```bash (package)
```

```bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update protocol public key of this candidate validator

```bash (package)
```

```bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network public key of this candidate validator

`bash (package)`

`bash (package)`

Update Narwhal worker public key of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update Narwhal worker public key of this candidate validator

`bash (package)`

`bash (package)`

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

`bash (package)`

`bash (package)`

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

`bash (package)`

`bash (package)`

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash

```

```bash

```

Function

Deactivate this validator's staking pool

```bash
 (package)
```

```bash
 (package)
```

```bash
 (package)
```

```bash
 (package)
```

Process pending stake and pending withdraws, and update the gas price.

```bash
 (package)
```

```bash
 (package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```bash
 (package)
```

```bash
 (package)
```

```bash
 (package)
```

```bash
 (package)
```

```bash
 (package)
```

```bash
 (package)
```

Request to add stake to the validator's staking pool at genesis

```bash
 (package)
```

```bash
 (package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```bash
 (package)
```

```bash
 (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
 (package)
```

```bash
 (package)
```

Set new gas price for the candidate validator.

```bash
 (package)
```

```bash
 (package)
```

Request to set new commission rate for the next epoch.

```bash
 (package)
```

```bash
 (package)
```

Set new commission rate for the candidate validator.

```bash
 (package)
```

```bash
 (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
 (package)
```

```bash
 (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
(package)
```

```bash
(package)
```

Process pending stake and pending withdraws, and update the gas price.

```bash
bash (package)
```

```bash
bash (package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Request to add stake to the validator's staking pool at genesis

```bash
bash (package)
```

```bash
bash (package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
bash (package)
```

```bash
bash (package)
```

Set new gas price for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new commission rate for the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Set new commission rate for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when

advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given [ValidatorMetadata](), called by both [new]() and new_for_testing.

```bash
```

```bash
```

Function

Process pending stake and pending withdraws, and update the gas price.

```bash
(package)
```

```bash
(package)
```

Request to add stake to the validator's staking pool, processed at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Request to add stake to the validator's staking pool at genesis

```bash
(package)
```

```bash
(package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
(package)
```

```bash
(package)
```

Set new gas price for the candidate validator.

```bash
(package)
```

```bash
(package)
```

Request to set new commission rate for the next epoch.

```bash
(package)
```

```bash
(package)
```

Set new commission rate for the candidate validator.

```bash
(package)
```

```bash
(package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Return the total amount staked with this validator

```bash

```

```bash

```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

`bash (package)`

`bash (package)`

Update name of the validator.

`bash (package)`

`bash (package)`

Update description of the validator.

`bash (package)`

`bash (package)`

Update image url of the validator.

`bash (package)`

`bash (package)`

Update project url of the validator.

`bash (package)`

`bash (package)`

Update network address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update network address of this candidate validator

`bash (package)`

`bash (package)`

Update p2p address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update p2p address of this candidate validator

`bash (package)`

`bash (package)`

Update primary address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

bash (package)

bash (package)

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash

```

```bash

```

Function

Request to add stake to the validator's staking pool, processed at the end of the epoch.

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

Request to add stake to the validator's staking pool at genesis

bash (package)

bash (package)

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

bash (package)

bash (package)

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

bash (package)

bash (package)

Set new gas price for the candidate validator.

bash (package)

bash (package)

Request to set new commission rate for the next epoch.

```
bash (package)
```

```
bash (package)
```

Set new commission rate for the candidate validator.

```
bash (package)
```

```
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

bash (package)

bash (package)

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```bash
bash (package)
```

```bash
bash (package)
```

Update name of the validator.

```bash
bash (package)
```

```bash (package)
```

Update description of the validator.

```bash (package)
```

```bash (package)
```

Update image url of the validator.

```bash (package)
```

```bash (package)
```

Update project url of the validator.

```bash (package)
```

```bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network address of this candidate validator

```bash (package)
```

```bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update p2p address of this candidate validator

```bash (package)
```

```bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update primary address of this candidate validator

```bash (package)
```

```bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update worker address of this candidate validator

```bash (package)
```

```bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash

```

```bash

```

Function

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Request to add stake to the validator's staking pool at genesis

```
bash (package)
```

```
bash (package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```
bash (package)
```

```
bash (package)
```

Set new gas price for the candidate validator.

```
bash (package)
```

```
bash (package)
```

Request to set new commission rate for the next epoch.

```
bash (package)
```

```
bash (package)
```

Set new commission rate for the candidate validator.

```
bash (package)
```

```
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Returns true if the validator is preactive.

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

```bash
bash (package)
```

```bash
bash (package)
```

Request to add stake to the validator's staking pool at genesis

```bash
bash (package)
```

```bash
bash (package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
bash (package)
```

```bash
bash (package)
```

Set new gas price for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new commission rate for the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Set new commission rate for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```bash (package)
```

```bash (package)
```

Update name of the validator.

```bash (package)
```

```bash (package)
```

Update description of the validator.

```bash (package)
```

```bash (package)
```

Update image url of the validator.

```bash (package)
```

```bash (package)
```

Update project url of the validator.

```bash (package)
```

```bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network address of this candidate validator

```bash (package)
```

```bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update p2p address of this candidate validator

```bash (package)
```

```bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update primary address of this candidate validator

```bash (package)
```

```bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```
```

```bash
```

```
```

Function

Request to add stake to the validator's staking pool at genesis

```
bash (package)
```

```
bash (package)
```

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```
bash (package)
```

```
bash (package)
```

Set new gas price for the candidate validator.

```
bash (package)
```

```
bash (package)
```

Request to set new commission rate for the next epoch.

```
bash (package)
```

```
bash (package)
```

Set new commission rate for the candidate validator.

```
bash (package)
```

```
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Returns true if the validator is preactive.

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```
bash (package)
```

```
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```
bash (package)
```

```
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

Request to withdraw stake from the validator's staking pool, processed at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```
bash (package)
```

```
bash (package)
```

Set new gas price for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new commission rate for the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Set new commission rate for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Return the total amount staked with this validator

```bash

```

```bash

```

Return the voting power of this validator.

```bash

```

```bash

```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash

```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```bash (package)
```

```bash (package)
```

Update description of the validator.

```bash (package)
```

```bash (package)
```

Update image url of the validator.

```bash (package)
```

```bash (package)
```

Update project url of the validator.

```bash (package)
```

```bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network address of this candidate validator

```bash (package)
```

```bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update p2p address of this candidate validator

```bash (package)
```

```bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update primary address of this candidate validator

```bash (package)
```

```bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update worker address of this candidate validator

```bash (package)
```

```bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

Request to set new gas price for the next epoch. Need to present a ValidatorOperationCap.

```bash
bash (package)
```

```bash
bash (package)
```

Set new gas price for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Request to set new commission rate for the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Set new commission rate for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash

```

Return the voting power of this validator.

```bash

```

```bash

```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

`bash (package)`

`bash (package)`

Update name of the validator.

`bash (package)`

`bash (package)`

Update description of the validator.

`bash (package)`

`bash (package)`

Update image url of the validator.

`bash (package)`

`bash (package)`

Update project url of the validator.

`bash (package)`

`bash (package)`

Update network address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update network address of this candidate validator

`bash (package)`

`bash (package)`

Update p2p address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update p2p address of this candidate validator

`bash (package)`

`bash (package)`

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

bash (package)

bash (package)

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash

```

```bash

```

Function

Set new gas price for the candidate validator.

bash (package)

bash (package)

Request to set new commission rate for the next epoch.

bash (package)

bash (package)

Set new commission rate for the candidate validator.

bash (package)

bash (package)

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

bash (package)

bash (package)

Process pending stakes and withdraws, called at the end of the epoch.

bash (package)

bash (package)

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

Request to set new commission rate for the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Set new commission rate for the candidate validator.

```bash
bash (package)
```

```bash
bash (package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```bash
(package)
```

Update project url of the validator.

```bash
(package)
```

```bash
(package)
```

Update network address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both [new](new) and new_for_testing.

```bash
```

```bash
```

Function

Set new commission rate for the candidate validator.

```bash
(package)
```

```bash
(package)
```

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```bash
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both [new](new) and new_for_testing.

```bash

```

```bash
```

```
```

Function

Deposit stakes rewards into the validator's staking pool, called at the end of the epoch.

```bash (package)
```

```bash (package)
```

Process pending stakes and withdraws, called at the end of the epoch.

```bash (package)
```

```bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Return the total amount staked with this validator

```bash

```

```bash

```

Return the voting power of this validator.

```bash

```

```bash

```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```bash
(package)
```

Update image url of the validator.

```bash
(package)
```

```bash
(package)
```

Update project url of the validator.

```bash
(package)
```

```bash
(package)
```

Update network address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

```
```

Function

Process pending stakes and withdraws, called at the end of the epoch.

```bash (package)
```

```bash (package)
```

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```bash
(package)
```

Update project url of the validator.

```bash
(package)
```

```bash
(package)
```

Update network address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

Returns true if the validator is preactive.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash

```
```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Return the total amount staked with this validator

```bash

```

```bash
```

```
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update p2p address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```
bash (package)
```
```
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```bash (package)
```

```bash (package)
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```bash (package)
```

```bash (package)
```

Update name of the validator.

```bash (package)
```

```bash (package)
```

Update description of the validator.

```bash (package)
```

```bash (package)
```

Update image url of the validator.

```bash (package)
```

```bash (package)
```

Update project url of the validator.

```bash (package)
```

```bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network address of this candidate validator

```bash (package)
```

```bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update p2p address of this candidate validator

```bash (package)
```

```bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```
bash (package)
```
```
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```bash (package)
```

```bash (package)
```

Update image url of the validator.

```bash (package)
```

```bash (package)
```

Update project url of the validator.

```bash (package)
```

```bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network address of this candidate validator

```bash (package)
```

```bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update p2p address of this candidate validator

```bash (package)
```

```bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update primary address of this candidate validator

```bash (package)
```

```bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update worker address of this candidate validator

```bash (package)
```

```bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```

```bash

```

Function

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```bash (package)
```

```bash (package)
```

Update description of the validator.

```bash (package)
```

```bash (package)
```

Update image url of the validator.

```bash (package)
```

```bash (package)
```

Update project url of the validator.

```bash (package)
```

```bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network address of this candidate validator

```bash (package)
```

```bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update p2p address of this candidate validator

```bash (package)
```

```bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update primary address of this candidate validator

```bash (package)
```

```bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update worker address of this candidate validator

```bash (package)
```

```bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
(package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash

```

```bash

```

Function

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update description of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network public key of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update Narwhal worker public key of this candidate validator

```
bash (package)
```

```
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```
bash (package)
```

```
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash

```

```bash

```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```
```bash
```

```
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both [new](new) and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

`bash (package)`

`bash (package)`

Update name of the validator.

`bash (package)`

`bash (package)`

Update description of the validator.

`bash (package)`

`bash (package)`

Update image url of the validator.

`bash (package)`

`bash (package)`

Update project url of the validator.

`bash (package)`

`bash (package)`

Update network address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update network address of this candidate validator

`bash (package)`

`bash (package)`

Update p2p address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update p2p address of this candidate validator

`bash (package)`

`bash (package)`

Update primary address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update primary address of this candidate validator

`bash (package)`

`bash (package)`

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash

```

```bash

```

```bash

```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash

```

```bash

```

Function

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```bash
(package)
```

```bash
(package)
```

Update network address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

`bash (package)`

`bash (package)`

Update name of the validator.

`bash (package)`

`bash (package)`

Update description of the validator.

`bash (package)`

`bash (package)`

Update image url of the validator.

`bash (package)`

`bash (package)`

Update project url of the validator.

`bash (package)`

`bash (package)`

Update network address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update network address of this candidate validator

`bash (package)`

`bash (package)`

Update p2p address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update p2p address of this candidate validator

`bash (package)`

`bash (package)`

Update primary address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update primary address of this candidate validator

`bash (package)`

`bash (package)`

Update worker address of this validator, taking effects from next epoch

`bash (package)`

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash

```

```bash

```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash

```

```bash

```

Function

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```
bash (package)
```
```
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
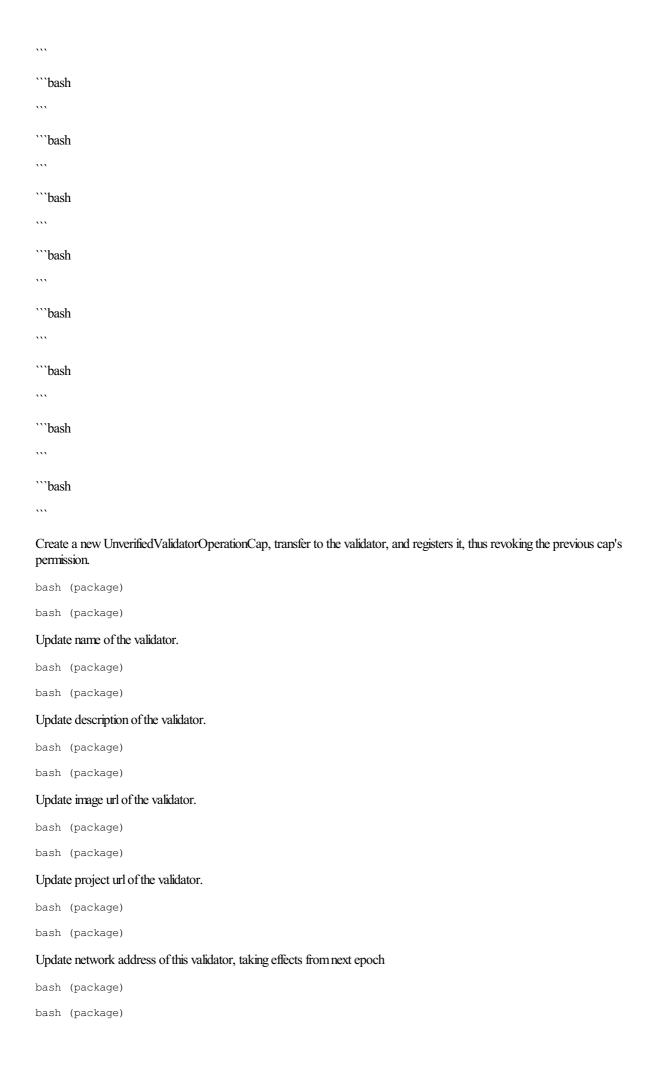(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

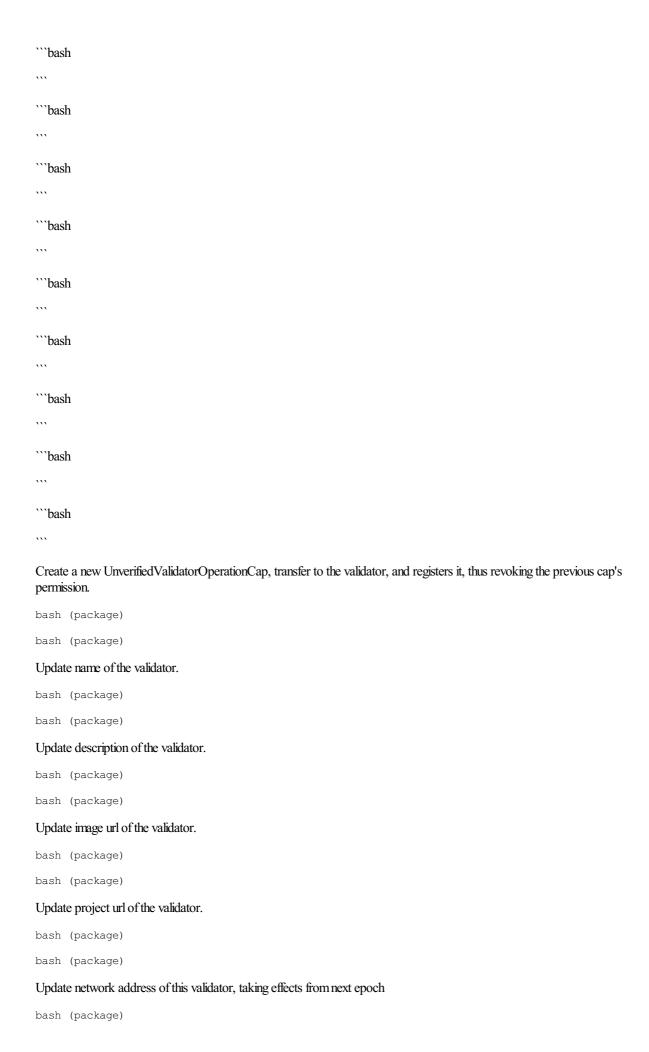Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given [ValidatorMetadata](), called by both [new]() and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```
bash (package)
```

```
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash

```

```bash

```

Function

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Return the total amount staked with this validator

```bash

```

```

```bash

```

Return the voting power of this validator.

```bash

```

```bash

```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

`bash (package)`

`bash (package)`

Update name of the validator.

`bash (package)`

`bash (package)`

Update description of the validator.

`bash (package)`

`bash (package)`

Update image url of the validator.

`bash (package)`

`bash (package)`

Update project url of the validator.

`bash (package)`

`bash (package)`

Update network address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update network address of this candidate validator

`bash (package)`

`bash (package)`

Update p2p address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update p2p address of this candidate validator

`bash (package)`

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Return the total amount staked with this validator

```bash

```

```bash

```

Return the voting power of this validator.

```bash

```

```bash

```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```bash
(package)
```

```bash
(package)
```

Update name of the validator.

```bash
(package)
```

```bash
(package)
```

Update description of the validator.

```bash
(package)
```

```bash
(package)
```

Update image url of the validator.

```bash
(package)
```

```bash
(package)
```

Update project url of the validator.

```bash
(package)
```

```bash
(package)
```

Update network address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash

```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash

```

```bash

```

Function

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update p2p address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update primary address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update worker address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update protocol public key of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network public key of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```bash
bash (package)
```

bash (package)

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```bash (package)```

```bash (package)```

Update name of the validator.

```bash (package)```

```bash (package)```

Update description of the validator.

```bash (package)```

```bash (package)```

Update image url of the validator.

```bash (package)```

```bash (package)```

Update project url of the validator.

```bash (package)```

```bash (package)```

Update network address of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update network address of this candidate validator

```bash (package)```

```bash (package)```

Update p2p address of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update p2p address of this candidate validator

```bash (package)```

```bash (package)```

Update primary address of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update primary address of this candidate validator

```bash (package)```

```bash (package)```

Update worker address of this validator, taking effects from next epoch

```bash (package)```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash

```

```bash

```

bash (package)

bash (package)

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash

```

```bash

```

Function

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

bash (package)

bash (package)

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
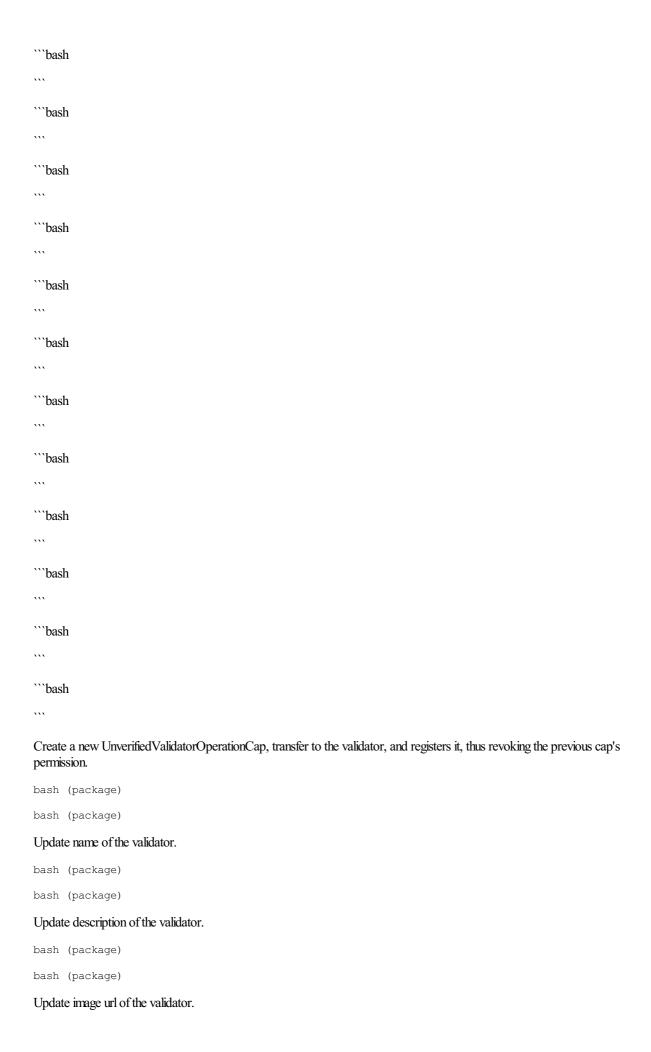bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given [ValidatorMetadata](), called by both [new]() and new_for_testing.

```bash

```

```bash

```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both [new](new) and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```
```

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

`bash (package)`

`bash (package)`

Update name of the validator.

`bash (package)`

`bash (package)`

Update description of the validator.

`bash (package)`

`bash (package)`

Update image url of the validator.

`bash (package)`

`bash (package)`

Update project url of the validator.

`bash (package)`

`bash (package)`

Update network address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update network address of this candidate validator

`bash (package)`

`bash (package)`

Update p2p address of this validator, taking effects from next epoch

`bash (package)`

```bash
(package)
```

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```
```bash
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

Return the total amount staked with this validator

```bash
```

```bash
```

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

bash (package)

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```bash (package)```

```bash (package)```

Update name of the validator.

```bash (package)```

```bash (package)```

Update description of the validator.

```bash (package)```

```bash (package)```

Update image url of the validator.

```bash (package)```

```bash (package)```

Update project url of the validator.

```bash (package)```

```bash (package)```

Update network address of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update network address of this candidate validator

```bash (package)```

```bash (package)```

Update p2p address of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update p2p address of this candidate validator

```bash (package)```

```bash (package)```

Update primary address of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update primary address of this candidate validator

```bash (package)```
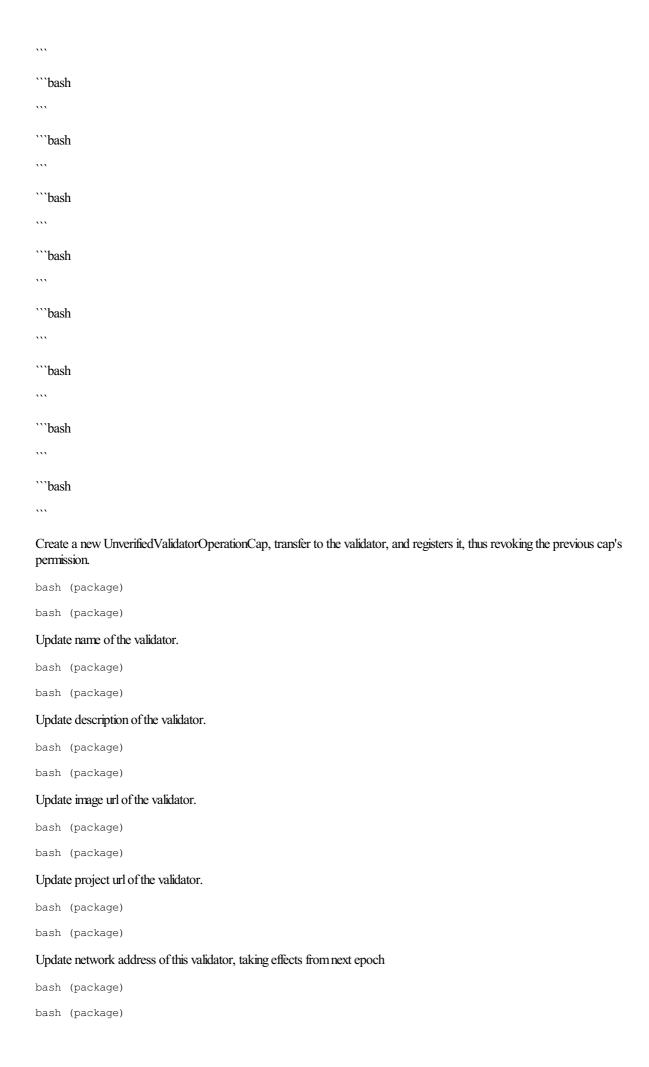
```bash (package)```

Update worker address of this validator, taking effects from next epoch

```bash (package)```

```bash (package)```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash
```

```
```

```bash
```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

Return the voting power of this validator.

```bash
```

```bash
```

Set the voting power of this validator, called only from validator_set.

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```
bash (package)
```

```
bash (package)
```

Update image url of the validator.

```
bash (package)
```

```
bash (package)
```

Update project url of the validator.

```
bash (package)
```

```
bash (package)
```

Update network address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

Set the voting power of this validator, called only from validator_set.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```
bash (package)
```

```
bash (package)
```

Update name of the validator.

```
bash (package)
```

```
bash (package)
```

Update description of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

bash (package)

bash (package)

Update name of the validator.

bash (package)

bash (package)

Update description of the validator.

bash (package)

bash (package)

Update image url of the validator.

bash (package)

bash (package)

Update project url of the validator.

bash (package)

bash (package)

Update network address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update network address of this candidate validator

bash (package)

bash (package)

Update p2p address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update p2p address of this candidate validator

bash (package)

bash (package)

Update primary address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update primary address of this candidate validator

bash (package)

bash (package)

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```

```

```bash

```

```bash

```

bash (package)

bash (package)

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash

```

```bash

```

Function

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

`bash (package)`

`bash (package)`

Update name of the validator.

`bash (package)`

`bash (package)`

Update description of the validator.

`bash (package)`

`bash (package)`

Update image url of the validator.

`bash (package)`

`bash (package)`

Update project url of the validator.

`bash (package)`

`bash (package)`

Update network address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update network address of this candidate validator

`bash (package)`

`bash (package)`

Update p2p address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update p2p address of this candidate validator

`bash (package)`

`bash (package)`

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

```

```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

`bash (package)`

`bash (package)`

Update name of the validator.

`bash (package)`

`bash (package)`

Update description of the validator.

`bash (package)`

`bash (package)`

Update image url of the validator.

`bash (package)`

`bash (package)`

Update project url of the validator.

`bash (package)`

`bash (package)`

Update network address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update network address of this candidate validator

`bash (package)`

`bash (package)`

Update p2p address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update p2p address of this candidate validator

`bash (package)`

`bash (package)`

Update primary address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update primary address of this candidate validator

```bash (package)
```

```bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update worker address of this candidate validator

```bash (package)
```

```bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update protocol public key of this candidate validator

```bash (package)
```

```bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network public key of this candidate validator

```bash (package)
```

```bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash (package)
```

```bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash (package)
```

```bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

`bash (package)`

`bash (package)`

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

`bash (package)`

`bash (package)`

Update name of the validator.

`bash (package)`

`bash (package)`

Update description of the validator.

`bash (package)`

`bash (package)`

Update image url of the validator.

`bash (package)`

`bash (package)`

Update project url of the validator.

`bash (package)`

`bash (package)`

Update network address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update network address of this candidate validator

`bash (package)`

`bash (package)`

Update p2p address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update p2p address of this candidate validator

`bash (package)`

`bash (package)`

Update primary address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash
```

```
```

```bash
```

```bash
```

```bash
```

bash (package)

bash (package)

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

bash (package)

```bash
bash (package)
```

Update name of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update description of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash

```

```bash

```

Function

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```bash
bash (package)
```

```bash
bash (package)
```

Update name of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update description of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network address of this candidate validator

```bash (package)
```

```bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update p2p address of this candidate validator

```bash (package)
```

```bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update primary address of this candidate validator

```bash (package)
```

```bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update worker address of this candidate validator

```bash (package)
```

```bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update protocol public key of this candidate validator

```bash (package)
```

```bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update network public key of this candidate validator

```bash (package)
```

```bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

```
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

`bash (package)`

`bash (package)`

Update name of the validator.

`bash (package)`

`bash (package)`

Update description of the validator.

`bash (package)`

`bash (package)`

Update image url of the validator.

`bash (package)`

`bash (package)`

Update project url of the validator.

`bash (package)`

`bash (package)`

Update network address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update network address of this candidate validator

`bash (package)`

`bash (package)`

Update p2p address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update p2p address of this candidate validator

`bash (package)`

`bash (package)`

Update primary address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```
```

```bash
```

```bash
```

```bash
```

bash (package)

bash (package)

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Macro function

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

bash (package)

bash (package)

Update name of the validator.

bash (package)

bash (package)

Update description of the validator.

bash (package)

bash (package)

Update image url of the validator.

bash (package)

bash (package)

Update project url of the validator.

bash (package)

bash (package)

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both [new](new) and new_for_testing.

```bash
```

```bash
```

Function

Create a new UnverifiedValidatorOperationCap, transfer to the validator, and registers it, thus revoking the previous cap's permission.

```bash
(package)
```

```bash
(package)
```

Update name of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update description of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update image url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update project url of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

Update name of the validator.

`bash (package)`

`bash (package)`

Update description of the validator.

`bash (package)`

`bash (package)`

Update image url of the validator.

`bash (package)`

`bash (package)`

Update project url of the validator.

`bash (package)`

`bash (package)`

Update network address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update network address of this candidate validator

`bash (package)`

`bash (package)`

Update p2p address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update p2p address of this candidate validator

`bash (package)`

`bash (package)`

Update primary address of this validator, taking effects from next epoch

`bash (package)`

`bash (package)`

Update primary address of this candidate validator

`bash (package)`

`bash (package)`

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash
```

```bash
```

```bash
```

bash (package)

bash (package)

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

Update description of the validator.

bash (package)

bash (package)

Update image url of the validator.

bash (package)

bash (package)

Update project url of the validator.

bash (package)

bash (package)

Update network address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update network address of this candidate validator

bash (package)

bash (package)

Update p2p address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update p2p address of this candidate validator

bash (package)

bash (package)

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

bash (package)

bash (package)

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

Update image url of the validator.

bash (package)

bash (package)

Update project url of the validator.

bash (package)

bash (package)

Update network address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update network address of this candidate validator

bash (package)

bash (package)

Update p2p address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update p2p address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

bash (package)

bash (package)

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

Update project url of the validator.

bash (package)

bash (package)

Update network address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update network address of this candidate validator

bash (package)

bash (package)

Update p2p address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update p2p address of this candidate validator

bash (package)

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

bash (package)

bash (package)

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

Update network address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update network address of this candidate validator

bash (package)

bash (package)

Update p2p address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update p2p address of this candidate validator

bash (package)

bash (package)

Update primary address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

bash (package)

bash (package)

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

Update network address of this candidate validator

bash (package)

bash (package)

Update p2p address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update p2p address of this candidate validator

bash (package)

bash (package)

Update primary address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update primary address of this candidate validator

bash (package)

bash (package)

Update worker address of this validator, taking effects from next epoch

bash (package)

bash (package)

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

Update p2p address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update p2p address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update primary address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update primary address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update worker address of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update protocol public key of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```
bash (package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash

```

```bash

```

Function

Update p2p address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update primary address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash

```

```bash

```

Function

Update primary address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update primary address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update worker address of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

Update primary address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when

advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given [ValidatorMetadata](), called by both [new]() and new_for_testing.

```bash
```

```bash
```

Function

Update worker address of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```
```

```bash
```

Function

Update worker address of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update protocol public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update network public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```
```

```bash
```

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

Update protocol public key of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update protocol public key of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update network public key of this candidate validator

```
bash (package)
```

```
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```
bash (package)
```

```
bash (package)
```

Update Narwhal worker public key of this candidate validator

```
bash (package)
```

```
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

Update protocol public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

Update network public key of this validator, taking effects from next epoch

```bash
bash (package)
```

```bash
bash (package)
```

Update network public key of this candidate validator

```bash
bash (package)
```

```bash
bash (package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

Update network public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash (package)
```

```bash (package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash (package)
```

```bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash (package)
```

```bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both [new](new) and new_for_testing.

```bash
```

```bash
```

Function

Update Narwhal worker public key of this validator, taking effects from next epoch

```bash (package)
```

```bash (package)
```

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```

Function

Update Narwhal worker public key of this candidate validator

```bash
(package)
```

```bash
(package)
```

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```bash
(package)
```

```bash
(package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash

```

```bash

```

```bash

```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash

```

```bash

```

Function

Effectutate all staged next epoch metadata for this validator. NOTE: this function SHOULD ONLY be called by validator_set when advancing an epoch.

```
bash (package)
```

```
bash (package)
```

Helper macro which extracts the value from Some and applies $f to it.

```bash

```

```bash

```

Aborts if validator metadata is valid

```bash

```

```bash
```

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Macro function

Helper macro which extracts the value from Some and applies $f to it.

```bash
```

```bash
```

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

```

## Function

Aborts if validator metadata is valid

```bash
```

```bash
```

```bash
```

```bash
```

bash (package)

bash (package)

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

```bash
```

```bash
```

bash (package)

bash (package)

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

bash (package)

bash (package)

Create a new validator from the given [ValidatorMetadata](#) , called by both [new](#) and new_for_testing.

```bash
```

```bash
```

Function

Create a new validator from the given ValidatorMetadata , called by both new and new_for_testing.

```bash
```

```bash
```