Split existing validator private keys

caution Charon is in a beta state and should be used with caution according to its Terms of Use .

This process should only be used if you want to split anexisting validator private key into multiple private key shares for use in a Distributed Validator Cluster. If your existing validator is not properly shut down before the Distributed Validator starts, your validator may be slashed.

If you are starting a new validator, you should follow a quickstart guide instead.

If you use MEV-Boost, make sure you turned off your MEV-Boost service for the time of splitting the keys, otherwise you may hitthis issue . Split an existing Ethereum validator key into multiple key shares for use in an Obol Distributed Validator Cluster .

Pre-requisites

- Ensure you have the existing validator keystores (the ones to split) and passwords.
- Ensure you havedocker
- installed.
- Make suredocker
- is running before executing the commands below.

Step 1. Clone the charon repo and copy existing keystore files

Clone thecharon repo.

Clone the repo

git clone https://github.com/ObolNetwork/charon.git

Change directory

cd charon/

Create a folder within this checked out repo

mkdir split_keys Copy the existing validatorkeystore.json files into this new folder. Alongside them, with a matching filename but ending with.txt should be the password to the keystore. E.g.,keystore-0.json keystore-0.txt

At the end of this process, you should have a tree like this:

├── split_keys | ├── keystore-0.json | ├── keystore-0.txt | ├── keystore-1.json | ├── keystore-1.txt | .. . | ├── keystore-*.json* | ├── *keystore*-.txt

Step 2. Split the keys using the charon docker command

Run the following docker command to split the keys:

CHARON_VERSION

E.g. v0.19.1

CLUSTER_NAME

The name of the cluster you want to create.

WITHDRAWAL_ADDRESS

The address you want to use for withdrawals.

FEE_RECIPIENT_ADDRESS

The address you want to use for fee payments.

NODES

The number of nodes in the cluster.

docker run --rm -v (pwd) :/opt/charon obolnetwork/charon: {CHARON_VERSION} create cluster --name = " {CLUSTER_NAME} " --withdrawal-addresses = " {WITHDRAWAL_ADDRESS} " --fee-recipient-addresses = " {FEE_RECIPIENT_ADDRESS} " --split-existing-keys --split-keys-dir = /opt/charon/split_keys --nodes {NODES} --network goerli The above command will createvalidator_keys along withcluster-lock.json in./cluster for each node.

Command output:

***** WARNING: Splitting keys ******* Please make sure any existing validator has been shut down for at least 2 finalised epochs before starting the charon cluster, otherwise slashing could occur.

Created charon cluster: --split-existing-keys = true

./cluster/ ├── node [0 -*] / Directory for each node | ├── charon-enr-private-key Charon networking private key for

node authentication | ├── cluster-lock.json Cluster lock defines the cluster lock file

which is signed by all nodes | ├── validator_keys Validator keystores and password | | ├── keystore-*.json *Validator private share key for duty signing* | | ├── *keystore*-.txt Keystore password files for keystore-*.json These split keys can now be used to start a charon cluster. Edit this page Previous Enable MEV Next Getting Started Monitoring your Node