

CLAIMS

1. A computer-implemented method to participate in a token distribution process for dividing a quantity of tokens at an input address associated with an input node into
5 a plurality of sub-quantities and distributing the sub-quantities to multiple output addresses associated with respective output nodes using a blockchain, the method comprising:

identifying an upstream node (U_i) and a plurality of downstream nodes (U_{ijk}) associated with a mixer node (U_{ij});
10 collaborating with the upstream node to generate a first commitment channel ($U_i \rightarrow U_{ij}$) for a first transaction between the upstream node and the mixer node; and

for each of the plurality of downstream nodes:
collaborating with the downstream node to generate a second
15 commitment channel ($U_{ij} \rightarrow U_{ijk}$) for a second transaction between the mixer node and the downstream node,
wherein an unlocking script for the first transaction is derived from an unlocking script for any one of the second transactions.
- 20 2. The method of claim 1, wherein generating the second commitment channel comprises:

collaborating with the downstream node to:
generate a first blockchain transaction (T_c) for committing a quantity
of tokens (x_i) to transfer to the downstream node;
25 generate a second blockchain transaction ($T_{r,0}$) that returns a committed quantity of tokens (x_i) to the mixer node; and

generate a third blockchain transaction (T_t) for executing transfer of the committed quantity of tokens (x_i) to the downstream node.

3. The method of claim 2, wherein the committed quantity of tokens (x_i) is determined based on a value allocation scheme for the mixer node which identifies a respective quantity to transfer to each of the plurality of downstream nodes.
- 5 4. The method of claim 2, wherein the third blockchain transaction (T_t) includes an unlocking script containing a second secret value (sv_{ijk}) associated with the second commitment channel.
5. The method of claim 4, further comprising obtaining the second secret value (sv_{ijk})
10 associated with the second commitment channel.
6. The method of claim 5, further comprising deriving a first secret value (sv_{ij}) associated with the first commitment channel using the second secret value (sv_{ijk}).
- 15 7. The method of claim 6, wherein the first secret value (sv_{ij}) is based on a secret key value (k_s) associated with the input node, the secret key value (k_s) being unknown to all mixer nodes.
8. The method of claim 1, further comprising transmitting a mixer key value (k_{ij}) to
20 the upstream node and to the input node.
9. The method of claim 1, wherein the mixer node includes a first address at which the mixer node detects transfer of a first quantity of tokens from the upstream node and a plurality of second addresses from which the mixer node transfers sub-quantities
25 of the first quantity to the downstream nodes.
10. The method of claim 9, wherein a total quantity of tokens at the plurality of second addresses is at least equal to the first quantity.
- 30 11. The method of claim 2, wherein the second blockchain transaction ($T_{r,0}$) includes a transaction parameter ($nLockTime$) specifying a point in time after which the second blockchain transaction becomes eligible for submission to the blockchain.

12. The method of claim 2, wherein the first blockchain transaction (T_c) is submitted to the blockchain after the second blockchain transaction ($T_{r,0}$) is signed by the downstream node.

5

13. The method of claim 1, wherein the first commitment channel is generated before any second commitment channel is generated.

14. A computing device to participate in a token distribution process including an input node, multiple output nodes, and a plurality of mixer nodes, the computing device comprising:

10

a processor;

memory;

a network interface to provide network connectivity; and

15

a blockchain application containing computer-executable instructions that, when executed by the processor, cause the processor to carry out the method claimed in claim 1.

15. A non-transitory processor-readable medium storing processor-executable instructions to participate in a token distribution process including an input node, multiple output nodes, and a plurality of mixer nodes, wherein the processor-executable instructions, when executed by a processor in one of the nodes, cause the processor to carry out the method claimed in claim 1.

20

25