

CLAIMS

1. A software architecture encoded on a non-transitory computer readable medium, the software architecture comprising:

a first protocol, wherein the first protocol is configured to form a plurality of groups, wherein each group of the plurality of groups comprises a set of randomly selected nodes from a network;

a creation protocol, wherein the creation protocol is configured to create a plurality of dealing messages, wherein each dealing message of the plurality of dealing messages comprises information from each dealer node in a selected group of the plurality of groups;

a fifth protocol, wherein the fifth protocol is configured to broadcast a plurality of blocks to a network, wherein each block of the plurality of blocks comprises a dealing message of the plurality of dealing messages;

a reading protocol, wherein the reading protocol is configured to read the dealing message to a receiver node of the selected group, wherein the receiver node is different from a dealer node whose information is contained in the corresponding dealing message

a verification protocol, wherein the verification protocol is configured to verify veracity of the corresponding dealing message, wherein the verification protocol is configured to be run by the receiver node;

a complaint protocol, wherein the complaint protocol is configured to generate a complaint message, wherein the complaint message is signed by the receiver node, and wherein the complaint protocol is configured to be run by the receiver node;

a justification protocol, wherein the justification protocol is configured to create a justification message, wherein the justification message comprises justifications to the complaint message, wherein the dealer node is configured to run the justification protocol to create the justification message; and

a registration protocol, wherein the registration protocol is configured to consolidate confirmation of a plurality of justified public keys, wherein each justified public key of the plurality of justified public keys corresponds to the each dealing message, wherein the verification protocol has accepted veracity of each justified public key of the plurality of justified public keys.

2. The software architecture of claim 1, wherein the creation protocol is configured to be run by each dealer node of the selected group.
3. The software architecture of claim 1, wherein the reading protocol is configured to be run by each receiver node of the selected group.
4. The software architecture of claim 1, further comprising:
a second protocol, wherein the second protocol is configured to select a first group from the plurality of groups formed by the first protocol.
5. The software architecture of claim 4, wherein the first group comprises the selected group.
6. The software architecture of claim 1, wherein the verification protocol is configured to verify an encrypted key share from the dealer node relative to a verification vector, wherein the information in the corresponding dealing message comprises the encrypted key share from the dealer node and the verification vector.
7. The software architecture of claim 6, wherein the verification protocol run by the receiver node is configured to verify the encrypted key share from the dealer node relative to the verification vector of the dealer node.
8. The software architecture of claim 6, wherein the complaint message comprises data regarding a lack of verification of the encrypted key share relative to the verification vector.
9. The software architecture of claim 1, wherein the verification protocol is configured to verify veracity of the justification message relative to the verification vector, wherein the verification protocol is configured to be run by a verifying node.
10. The software architecture of claim 9, wherein the verification protocol rejects veracity of the justification message.
11. The software architecture of claim 9, wherein the verification protocol accepts veracity of the justification message.
12. The software architecture of claim 1, wherein the registration protocol is run by the verifying node.

13. The software architecture of claim 1, further comprising a second protocol, wherein the second protocol is configured to randomly select a notary group from the plurality of groups, wherein the notary group is configured to use the fourth protocol.
14. The software architecture of claim 1, further comprising a fifth protocol, wherein the fifth protocol is configured to broadcast a plurality of blocks to a network, wherein each block of the plurality of blocks comprises a dealing message of the plurality of dealing messages.
15. The software architecture of claim 14, further comprising a fourth protocol, wherein the fourth protocol is configured to notarize the each block of the plurality of blocks.