CLAIMS

What is claimed is:

1. A method for providing data privacy in a private distributed ledger, comprising:

a distributed application preparing a transaction payload for a private transaction;

the distributed application sending a payload storage message comprising a hash digest, an encrypted payload, and an encrypted symmetric key to a transaction key manager;

the transaction key manager storing the hash and the encrypted payload;

the distributed application sending a pending transaction comprising the hash digest to a first node in a plurality of nodes; and

one of the plurality of nodes proposing a block comprising the pending transaction to the other nodes.

2. The method of claim 1, wherein the step of preparing a transaction payload for a private transaction comprises:

generating the symmetric key;

encrypting the payload with the symmetric key;

calculating the hash digest of the encrypted payload; and

encrypting the symmetric key with public keys of parties to the transaction.

3. The method of claim 1, further comprising:

each node validating the pending transaction in the block.

4. The method of claim 3, wherein the step of validating the pending transaction in the block comprises:

identifying the pending transaction as a private transaction.

5. The method of claim 4, wherein the pending transaction is identified as private by a sentinel byte.

6. The method of claim 4, wherein the pending transaction is identified as private by a flag in a body of the pending transaction.

7. The method of claim 6, further comprising:

the node sending a transaction payload request message to a transaction key manager associated with the node.

8. The method of claim 7, wherein the transaction payload request message comprises the node's public key, the hash digest, and the node's signature.

9. The method of claim 1, further comprising:

the transaction key manager returning a decrypted payload and encrypted symmetric key to the node in response to the node being a party to the transaction.

10. The method of claim 1, further comprising:

the transaction key manager returning an indication to the node that the transaction is private in response to the node not being a party to the transaction.

11. A distributed ledger network comprising:

a first node comprising a first node computer processor and hosting a central ledger comprising a plurality of entries for public transactions and private transactions, wherein the entries for public transactions comprise transaction payloads for the respective public transaction, and the entries for private transactions comprise a cryptographic hash digest of a transaction payload for the respective private transaction; and

a plurality of second nodes each comprising a second node computer processor and hosting a second ledger comprising the public transactions and private transaction hash digests for transactions to which the node is not a party, and a private database comprising the private state derived from private transactions to which the node is a party.

12. The distributed ledger network of claim 11, wherein each second node further comprises a transaction key manager that stores transaction payloads that are private to the respective nodes.

13. The distributed ledger network of claim 11, wherein the public databases for each second node are identical.

14. A method for providing data privacy in a private distributed ledger, comprising:

a node in a distributed ledger network receiving a block comprising a first transaction and a second transaction;

a node computer processor determining that the first transaction is a private transaction;

the node computer processor executing contract code associated with the first transaction;

the node computer processor determining that the node is not a party to the second transaction; and

the node skipping executing of contract code associated with the second transaction.

15. The method of claim 14, wherein the node computer processor determines that the first transaction or second transaction is a private transaction based on a flag in a body of the first or second transaction.

16. The method of claim 14, further comprising:

the node computer processor communicating a node public key and the transaction hash digest for the first transaction to a transaction key manager for the node; and

the node computer processor receiving an encrypted transaction payload for the first transaction and decrypting the encrypted transaction payload, the transaction payload comprising the contract code associated with the first transaction in response to the node being a party to the first transaction.

17. The method of claim 14, further comprising:

the node computer processor communicating a node public key and the transaction hash digest for the first transaction to a transaction key manager for the node; and

the node computer processor receiving an indication that the node is not a party to the first transaction in response to the node not being a party to the first transaction.

18. The method of claim 14, further comprising:

the node computer processor validating the block.

19. The method of claim 18, wherein the step of validating the block comprises validation of the public state, wherein each node has a different private state.

20. The method of claim 14, wherein the distributed ledger comprises private transactions and public transactions, and

wherein the private transactions and the public transactions are visible and cryptographically immutable.