

CLAIMS

1. A computer-implemented method, comprising:

receiving, by a first blockchain node, an encrypted transaction comprising a smart contract that includes code, wherein the code of the smart contract comprises a contract state indicated by a privacy identifier;

decrypting, by the first blockchain node, the encrypted transaction to obtain the code of the smart contract in plaintext;

executing, by the first blockchain node, the code of the smart contract in plaintext in a trusted execution environment;

encrypting, by the first blockchain node using a key, the contract state indicated by the privacy identifier; and

writing, by the first blockchain node, the encrypted contract state indicated by the privacy identifier to a database.

2. The computer-implemented method of claim 1, wherein the smart contract of the encrypted transaction received by the first blockchain node comprises one of:

a smart contract written by a user in a high-level language on the first blockchain node;

a smart contract written by a user in a high-level language on a client device; or

a smart contract in a transaction sent by the client device by using a second blockchain node.

3. The computer-implemented method of claim 1, wherein the smart contract is originally written in a high-level language, and is compiled using a compiler into bytecode before execution.

4. The computer-implemented method of claim 3, wherein the smart contract in the high-level language comprises a first privacy identifier indicating a contract state that requires privacy protection, and the smart contract in the bytecode comprises a second privacy identifier indicating a contract state that requires privacy protection; and

the first privacy identifier is the same as or has a mapping relationship with the second privacy identifier.

5. The computer-implemented method of claim 1, wherein for the encrypted transaction that comprises the smart contract and that is received by the first blockchain node, an encryption process uses a symmetric encryption method, an asymmetric encryption method, or a combination of a symmetric encryption method and an asymmetric encryption method.

6. The computer-implemented method of claim 5, wherein
a private key for symmetric encryption is used to encrypt the transaction comprising the smart contract, and the first blockchain node uses the private key for symmetric encryption to decrypt the transaction to obtain the code of the smart contract; or
a public key for asymmetric encryption is used to encrypt the transaction comprising the smart contract, and the first blockchain node uses a corresponding private key for asymmetric encryption to decrypt the transaction to obtain the code of the smart contract.

7. The computer-implemented method of claim 5, wherein the combination of a symmetric encryption method and an asymmetric encryption method comprises a digital envelope encryption method.

8. The computer-implemented method of claim 1, wherein the key comprises a key for symmetric encryption or a key for asymmetric encryption.

9. A non-transitory, computer-readable medium storing instructions executable by a computer system that upon execution cause the computer system to perform operations comprising:

receiving, by a first blockchain node, an encrypted transaction comprising a smart contract that includes code, wherein the code of the smart contract comprises a contract state indicated by a privacy identifier;

decrypting, by the first blockchain node, the encrypted transaction to obtain the code of the smart contract in plaintext;

executing, by the first blockchain node, the code of the smart contract in plaintext in a trusted execution environment;

encrypting, by the first blockchain node using a key, the contract state indicated by the privacy identifier; and

writing, by the first blockchain node, the encrypted contract state indicated by the privacy identifier to a database.

10. The non-transitory, computer-readable medium of claim 9, wherein the smart contract of the encrypted transaction received by the first blockchain node comprises one of:

a smart contract written by a user in a high-level language on the first blockchain node;

a smart contract written by a user in a high-level language on a client device; or

a smart contract in a transaction sent by the client device by using a second blockchain node.

11. The non-transitory, computer-readable medium of claim 9, wherein the smart contract is originally written in a high-level language, and is compiled using a compiler into bytecode before execution.

12. The non-transitory, computer-readable medium of claim 11, wherein the smart contract in the high-level language comprises a first privacy identifier indicating a contract state that requires privacy protection, and the smart contract in the bytecode comprises a second privacy identifier indicating a contract state that requires privacy protection; and

the first privacy identifier is the same as or has a mapping relationship with the second privacy identifier.

13. The non-transitory, computer-readable medium of claim 9, wherein for the encrypted transaction that comprises the smart contract and that is received by the first blockchain node, an encryption process uses a symmetric encryption method, an asymmetric encryption method, or a combination of a symmetric encryption method and an asymmetric encryption method.

14. The non-transitory, computer-readable medium of claim 13, wherein

a private key for symmetric encryption is used to encrypt the transaction comprising the smart contract, and the first blockchain node uses the private key for symmetric encryption to decrypt the transaction to obtain the code of the smart contract; or

a public key for asymmetric encryption is used to encrypt the transaction comprising the smart contract, and the first blockchain node uses a corresponding private key for asymmetric encryption to decrypt the transaction to obtain the code of the smart contract.

15. The non-transitory, computer-readable medium of claim 13, wherein the combination of a symmetric encryption method and an asymmetric encryption method comprises a digital envelope encryption method.

16. The non-transitory, computer-readable medium of claim 9, wherein the key comprises a key for symmetric encryption or a key for asymmetric encryption.

17. A computer-implemented system, comprising:

one or more computers; and

one or more computer memory devices interoperably coupled with the one or more computers and having tangible, non-transitory, machine-readable media storing instructions that, when executed by the one or more computers, cause the one or more computers to perform operations comprising:

receiving, by a first blockchain node, an encrypted transaction comprising a smart contract, wherein code of the smart contract comprises a contract state indicated by a privacy identifier;

decrypting, by the first blockchain node, the encrypted transaction to obtain the code of the smart contract in plaintext;

executing, by the first blockchain node, the code of the smart contract in plaintext in a trusted execution environment;

encrypting, by the first blockchain node using a key, the contract state indicated by the privacy identifier; and

writing, by the first blockchain node, the encrypted contract state indicated by the privacy identifier to a database.

18. The computer-implemented system of claim 17, wherein the smart contract of the encrypted transaction received by the first blockchain node comprises one of:

a smart contract written by a user in a high-level language on the first blockchain node;

a smart contract written by a user in a high-level language on a client device; or

a smart contract in a transaction sent by the client device by using a second blockchain node.

19. The computer-implemented system of claim 17, wherein the smart contract is originally written in a high-level language, and is compiled using a compiler into bytecode before execution.

20. The computer-implemented system of claim 19, wherein the smart contract in high-level language comprises a first privacy identifier indicating a contract state that requires privacy protection, and the smart contract in the bytecode comprises a second privacy identifier indicating a contract state that requires privacy protection; and

the first privacy identifier is the same as or has a mapping relationship with the second privacy identifier.