CLAIMS

1. A method for deploying a smart contract, comprising:

sending, by a client device, a bytecode of the smart contract to a first trusted execution environment TEE, and receiving a machine code obtained by Ahead of Time AoT compilation of the bytecode by the first TEE;

receiving, by a blockchain node, a transaction for creating the smart contract submitted by the client device, wherein the transaction comprises the machine code of the smart contract; and

completing, by the blockchain node, a deployment of the smart contract in response to determining that the machine code is obtained by compiling the bytecode by a trusted TEE,

wherein the blockchain node determining that the machine code is obtained by compiling the bytecode by the trusted TEE comprises:

obtaining, by the blockchain node, a signature of the machine code by the first TEE by using a private key from the transaction; and

in response to determining that the signature is successfully verified by a public key corresponding to a predefined trusted TEE mirror image, the blockchain node determining that the first TEE is the trusted TEE.

2. The method of claim 1, wherein compiling the bytecode of the smart contract into the machine code by the AoT compilation comprises:

performing optimizing compilation in a process of the AoT compilation of the bytecode of the smart contract.

3. The method of claim 1, wherein the first TEE is deployed on any one of the following: the client device, any blockchain node in a blockchain network, and a third-party server different from the client device and the blockchain node.

4. The method of claim 1, wherein the private key is distributed to the first TEE by a key management server in response to determining that the first TEE passes remote certification.

5. The method of claim 1, wherein the trusted TEE mirror image comprises: a mirror image of a trusted enclave based on software guard protection SGX.

6. The method of claim 1, wherein the public key corresponding to the trusted TEE mirror image is recorded in a system contract; the method further comprises:

invoking, by the blockchain node, the system contract, and passing the signature in the system contract; and

receiving, by the blockchain node, an identification result returned by the system contract, wherein the identification result is used to indicate whether the signature is successfully verified by the public key corresponding to the trusted TEE mirror image.

7. The method of claim 6, wherein the blockchain node invoking the system contract comprises:

reading, by the blockchain node, a contract address of the system contract from the transaction, and invoking the system contract based on the contract address; or

invoking, by the blockchain node, the system contract based on the contract address of the system contract defined in a chain code in response to determining that a transaction type is a contract deployment type.

8. The method of claim 7, wherein the system contract is deployed in a genesis block; or, a management authority of the system contract is a blockchain administrator.

9. A method for deploying a smart contract, comprising:

receiving, by a blockchain node, a transaction for creating a smart contract, wherein the transaction comprises a machine code of the smart contract, and the machine code is obtained by Ahead of Time AoT compilation of a bytecode of the smart contract by a first trusted execution environment TEE; and

completing, by the blockchain node, a deployment of the smart contract in response to determining that the machine code is obtained by compiling the bytecode by a trusted TEE,

wherein the blockchain node determining that the machine code is obtained by compiling the bytecode by the trusted TEE comprises:

obtaining, by the blockchain node, a signature of the machine code by the first TEE by using a private key from the transaction; and

in response to determining that the signature is successfully verified by a public key corresponding to a predefined trusted TEE mirror image, the blockchain node determining that the first TEE is the trusted TEE.

10. The method of claim 9, wherein compiling the bytecode of the smart contract into the machine code by the AoT compilation comprises:

performing optimizing compilation in a process of the AoT compilation of the bytecode of the smart contract.

11. The method of claim 9, wherein the first TEE is deployed on any one of the following: a client device that submits the transaction, any blockchain node in a blockchain network, and a third-party server different from the client device and the blockchain node.

12. The method of claim 9, wherein the private key is distributed to the first TEE by a key management server in response to determining that the first TEE passes remote certification.

13. The method of claim 9, wherein the trusted TEE mirror image comprises: a mirror image of a trusted enclave based on software guard protection SGX.

14. The method of claim 9, wherein the public key corresponding to the trusted TEE mirror image is recorded in a system contract; the method further comprises:

invoking, by the blockchain node, the system contract, and passing the signature in the system contract; and

receiving, by the blockchain node, an identification result returned by the system contract, wherein the identification result is used to indicate whether the signature is successfully verified by the public key corresponding to the trusted TEE mirror image.

15. The method of claim 14, wherein the blockchain node invoking the system contract comprises:

reading, by the blockchain node, a contract address of the system contract from the transaction, and invoking the system contract based on the contract address; or

invoking, by the blockchain node, the system contract based on the contract address of the system contract defined in a chain code in response to determining that a transaction type is a contract deployment type.

16. The method of claim 15, wherein the system contract is deployed in a genesis block; or, a management authority of the system contract is a blockchain administrator.

17. A blockchain node for deploying a smart contract, wherein the method described in any one of claims 1-8 and 9-16 is performed.

18. A blockchain node for deploying a smart contract, comprising:

a processor, and

a memory storing a program thereon,

wherein the program is executed by the processor to perform the method according to any one of claims 1-8 and 9-16.

19. A storage medium for storing a program, wherein the program is executed to perform an operation of any one of claims 1-8 and 9-16.