

CLAIMS

1. A method for preserving data integrity when integrating secure multiparty computation (SMPC) and blockchain technology, the method comprising:
 - splitting, via a data publisher, data into a plurality of data secret shares using a SMPC protocol, wherein each data secret share of the plurality of data secret shares is assigned to an SMPC compute node of a plurality of SMPC compute nodes; and
 - for each respective data secret share:
 - identifying, from the plurality of SMPC compute nodes, a respective SMPC compute node assigned to the respective data secret share;
 - retrieving a respective public key of the respective SMPC compute node;
 - encrypting the respective data secret share using the respective public key; and
 - publishing the encrypted data respective secret share on a blockchain.
2. The method of claim 1, wherein the plurality of SMPC compute nodes are members of a blockchain network managing the blockchain.
3. The method of claim 1, wherein at least one SMPC compute node of the plurality of SMPC compute nodes is not a member of a blockchain network managing the blockchain.
4. The method of claim 1, further comprising publishing the encrypted respective secret share on a database system.
5. The method of claim 4, further comprising publishing a signature of the encrypted respective secret share on the blockchain.
6. The method of claim 1, further comprising:
 - publishing the respective secret share to a respective database system of a plurality of database systems such that no single entity has access to multiple secret shares in one database system.

7. The method of claim 1, further comprising:
 - generating, via the data publisher, a random number (R) within a public mod value (M) known by the plurality of SMPC compute nodes;
 - computing $V \times R \times \text{MAC}_{\text{key}2}$ to determine $\text{MAC}_{v2} \bmod M$, wherein V is the data, $\text{MAC}_{\text{key}2}$ is a message authentication code key known by the plurality of SMPC compute nodes and MAC_{v2} is a message authentication code;
 - determining R^{-1} , wherein R^{-1} is an inverse of R in mod M; and
 - including additional secret shares of R^{-1} and MAC_{v2} in a plurality of MAC secret shares for encryption and publication on the blockchain, wherein each respective secret share of R^{-1} and MAC_{v2} is matched with a corresponding data secret share in the plurality of data secret shares.

8. The method of claim 7, further comprising for each respective data secret share and each respective MAC secret share:
 - retrieving, via a data querier, the encrypted respective data secret share and the respective MAC secret share from the blockchain;
 - sending the encrypted respective data secret share and the respective MAC secret share to the assigned respective SMPC compute node and an indication of a secure function to perform, wherein the respective SMPC compute node is configured to:
 - decrypt the encrypted respective data secret share using a private key of the respective SMPC compute node;
 - convert the $\text{MAC}_{\text{key}2}$ of the respective MAC secret share to $\text{MAC}_{\text{key}1}$ by computing $\text{MAC}_{v2} \times F \times R^{-1} \bmod M$;
 - in response to verifying data integrity of the decrypted respective data secret share, executing the secure function on the decrypted respective data secret share;
 - convert the $\text{MAC}_{\text{key}1}$ to the $\text{MAC}_{\text{key}2}$ of the respective MAC secret share by computing $\text{MAC}_{v1} \times R \times F^{-1} \bmod M$;
 - publish, on the blockchain, a digital signature of a result of the execution; and
 - transmit the result to the data querier.

9. The method of claim 8, further comprising:

receiving, via the data querier, a respective result from each of the plurality of SMPC compute nodes and the plurality of MAC secret shares;

joining each of the plurality of MAC secret shares and the respective results received from the plurality of SMPC compute nodes to determine V , R^{-1} and MAC_{v2} ;

verifying the data integrity by determining whether $V \times R \times MAC_{key2}$ equals $MAC_{v2} \bmod M$; and

in response to determining that $V \times R \times MAC_{key2}$ equals $MAC_{v2} \bmod M$, determining the respective results are authentic.

10. The method of claim 8, further comprising verifying the result via an auditor that is configured to:

request the decrypted data secret share and the secure function;

perform the secure function on the decrypted data secret share to determine a second result; and

compare the result of the respective SMPC compute node with the second result.

11. The method of claim 8, further comprising verifying the result via an auditor that is configured to:

transfer the plurality of data secret shares into a different secure environment;

perform the secure function within the different secure environment to determine a second result; and

compare the result of the respective SMPC compute node with the second result.

12. The method of claim 1, wherein a data querier transmits assignment information to each of the plurality of SMPC compute nodes indicating which respective encrypted data secret share to retrieve from the blockchain, and wherein each respective SMPC compute node directly retrieves the assigned respective secret share from the blockchain.

13. The method of claim 1, wherein a respective SMPC compute node of the plurality of SMPC compute nodes detects the publishing of new data into the blockchain and automatically retrieves and decrypts the encrypted respective data secret share.

14. A system for preserving data integrity when integrating secure multiparty computation (SMPC) and blockchain technology, the system comprising:
a hardware processor configured to:

split, via a data publisher, data into a plurality of data secret shares using a SMPC protocol, wherein each data secret share of the plurality of data secret shares is assigned to an SMPC compute node of a plurality of SMPC compute nodes; and

for each respective data secret share:

identify, from the plurality of SMPC compute nodes, a respective SMPC compute node assigned to the respective data secret share;

retrieve a respective public key of the respective SMPC compute node;

encrypt the respective data secret share using the respective public key; and

publish the encrypted data respective secret share on a blockchain.

15. The system of claim 14, wherein the plurality of SMPC compute nodes are members of a blockchain network managing the blockchain.

16. The system of claim 14, wherein at least one SMPC compute node of the plurality of SMPC compute nodes is not a member of a blockchain network managing the blockchain.

17. The system of claim 14, wherein the hardware processor is further configured to publish the encrypted respective secret share on a database system.

18. The system of claim 17, wherein the hardware processor is further configured to publish a signature of the encrypted respective secret share on the blockchain.

19. The system of claim 14, wherein the hardware processor is further configured to:

publish the respective secret share to a respective database system of a plurality of database systems such that no single entity has access to multiple secret shares in one database system.

20. The system of claim 14, wherein the hardware processor is further configured to:

generate, via the data publisher, a random number (R) within a public mod value (M) known by the plurality of SMPC compute nodes;

compute $V \times R \times \text{MAC}_{\text{key}2}$ to determine $\text{MAC}_{v2} \bmod M$, wherein V is the data, $\text{MAC}_{\text{key}2}$ is a message authentication code key known by the plurality of SMPC compute nodes and MAC_{v2} is a message authentication code;

determine R^{-1} , wherein R^{-1} is an inverse of R in mod M; and

include additional secret shares of R^{-1} and MAC_{v2} in a plurality of MAC secret shares for encryption and publication on the blockchain, wherein each respective secret share of R^{-1} and MAC_{v2} is matched with a corresponding data secret share in the plurality of data secret shares.

21. A non-transitory computer readable medium storing thereon computer executable instructions for preserving data integrity when integrating secure multiparty computation (SMPC) and blockchain technology, including instructions for:

splitting, via a data publisher, data into a plurality of data secret shares using a SMPC protocol, wherein each data secret share of the plurality of data secret shares is assigned to an SMPC compute node of a plurality of SMPC compute nodes; and

for each respective data secret share:

identifying, from the plurality of SMPC compute nodes, a respective SMPC compute node assigned to the respective data secret share;

retrieving a respective public key of the respective SMPC compute node;

encrypting the respective data secret share using the respective public key; and

publishing the encrypted data respective secret share on a blockchain.

CLAIMS

1. A method for preserving data integrity when integrating secure multiparty computation (SMPC) and blockchain technology, the method comprising:
 - splitting, via a data publisher, data into a plurality of data secret shares using a SMPC protocol;
 - generating, using the SMPC protocol, a plurality of message authentication code (MAC) secret shares of MAC condition parameters that ensure integrity of the data by:
 - generating, via the data publisher, a random number within a public mod value known by a plurality of SMPC compute nodes;
 - computing a message authentication code (MAC_{v2}) based on the data, the random number, and a message authentication code key known by the plurality of SMPC compute nodes;
 - determining an inverse of the random number; and
 - including additional secret shares of the inverse and the MAC_{v2} in the plurality of MAC secret shares for encryption and publication on the blockchain, wherein each respective secret share of the inverse and the MAC_{v2} is matched with a corresponding data secret share in the plurality of data secret shares;
 - wherein each data secret share of the plurality of data secret shares and each MAC secret share of the plurality of MAC secret shares is assigned to an SMPC compute node of the plurality of SMPC compute nodes; and
 - for each respective data secret share and MAC secret share:
 - identifying, from the plurality of SMPC compute nodes, a respective SMPC compute node assigned to the respective data secret share and MAC secret share;
 - retrieving a respective public key of the respective SMPC compute node;
 - encrypting the respective data secret share and MAC secret share using the respective public key; and
 - publishing the encrypted respective data secret share and MAC secret share on a blockchain.
2. The method of claim 1, wherein the plurality of SMPC compute nodes are members of a blockchain network managing the blockchain.

3. The method of claim 1, wherein at least one SMPC compute node of the plurality of SMPC compute nodes is not a member of a blockchain network managing the blockchain.
4. The method of claim 1, further comprising publishing the encrypted respective data secret share and MAC secret share on a database system.
5. The method of claim 4, further comprising publishing a signature of the encrypted respective data secret share and MAC secret share on the blockchain.
6. The method of claim 1, further comprising:
publishing the encrypted respective data secret share and MAC secret share to a respective database system of a plurality of database systems such that no single entity has access to multiple data secret shares and multiple MAC secret shares in one database system.
7. The method of claim 1, wherein computing MAC_{v2} comprises:
computing $V \times R \times MAC_{key2}$ to determine $MAC_{v2} \bmod M$, wherein V is the data, MAC_{key2} is the message authentication code key known by the plurality of SMPC compute nodes, R is the random number within the public mod value (M), and R^{-1} is the inverse of R in mod M .
8. The method of claim 7, further comprising for each respective data secret share and each respective MAC secret share:
retrieving, via a data querier, the encrypted respective data secret share and the respective MAC secret share from the blockchain;
sending the encrypted respective data secret share and the respective MAC secret share to the assigned respective SMPC compute node and an indication of a secure function to perform, wherein the respective SMPC compute node is configured to:
decrypt the encrypted respective data secret share using a private key of the respective SMPC compute node;
convert the MAC_{key2} of the respective MAC secret share to MAC_{key1} by computing $MAC_{v2} \times F \times R^{-1} \bmod M$;

in response to verifying data integrity of the decrypted respective data secret share, executing the secure function on the decrypted respective data secret share;

convert the MAC_{key1} to the MAC_{key2} of the respective MAC secret share by computing $MAC_{v1} \times R \times F^{-1} \bmod M$;

publish, on the blockchain, a digital signature of a result of the execution; and

transmit the result to the data querier.

9. The method of claim 8, further comprising:

receiving, via the data querier, a respective result from each of the plurality of SMPC compute nodes and the plurality of MAC secret shares;

joining each of the plurality of MAC secret shares and the respective results received from the plurality of SMPC compute nodes to determine V , R^{-1} and MAC_{v2} ;

verifying the data integrity by determining whether $V \times R \times MAC_{key2}$ equals $MAC_{v2} \bmod M$; and

in response to determining that $V \times R \times MAC_{key2}$ equals $MAC_{v2} \bmod M$, determining the respective results are authentic.

10. The method of claim 8, further comprising verifying the result via an auditor that is configured to:

request the decrypted data secret share and the secure function;

perform the secure function on the decrypted data secret share to determine a second result;

and

compare the respective result of the respective SMPC compute node with the second result.

11. The method of claim 8, further comprising verifying the result via an auditor that is configured to:

transfer the plurality of data secret shares into a different secure environment;

perform the secure function within the different secure environment to determine a second result; and

compare the respective result of the respective SMPC compute node with the second result.

12. The method of claim 1, wherein a data querier transmits assignment information to each of the plurality of SMPC compute nodes indicating which respective encrypted data secret share to retrieve from the blockchain, and wherein each respective SMPC compute node directly retrieves the assigned respective secret share from the blockchain.

13. The method of claim 1, wherein a respective SMPC compute node of the plurality of SMPC compute nodes detects the publishing of new data into the blockchain and automatically retrieves and decrypts the encrypted respective data secret share.

14. A system for preserving data integrity when integrating secure multiparty computation (SMPC) and blockchain technology, the system comprising:

a hardware processor configured to:

split, via a data publisher, data into a plurality of data secret shares using a SMPC protocol;

generate, using the SMPC protocol, a plurality of message authentication code (MAC)

secret shares of MAC condition parameters that ensure integrity of the data by:

generating, via the data publisher, a random number within a public mod value known by a plurality of SMPC compute nodes;

computing a message authentication code (MAC_{v2}) based on the data, the random number, and a message authentication code key known by the plurality of SMPC compute nodes;

determining an inverse of the random number; and

including additional secret shares of the inverse and the MAC_{v2} in the plurality of MAC secret shares for encryption and publication on the blockchain, wherein each respective secret share of the inverse and the MAC_{v2} is matched with a corresponding data secret share in the plurality of data secret shares;

wherein each data secret share of the plurality of data secret shares and each MAC secret share of the plurality of MAC secret shares is assigned to an SMPC compute node of the plurality of SMPC compute nodes; and

for each respective data secret share and MAC secret share:

identify, from the plurality of SMPC compute nodes, a respective SMPC compute node assigned to the respective data secret share and MAC secret share;

retrieve a respective public key of the respective SMPC compute node;
encrypt the respective data secret share and MAC secret share using the respective public key; and
publish the encrypted respective data secret share and MAC secret share on a blockchain.

15. The system of claim 14, wherein the plurality of SMPC compute nodes are members of a blockchain network managing the blockchain.

16. The system of claim 14, wherein at least one SMPC compute node of the plurality of SMPC compute nodes is not a member of a blockchain network managing the blockchain.

17. The system of claim 14, wherein the hardware processor is further configured to publish the encrypted respective data secret share and MAC secret share on a database system.

18. The system of claim 17, wherein the hardware processor is further configured to publish a signature of the encrypted respective data secret share and MAC secret share on the blockchain.

19. The system of claim 14, wherein the hardware processor is further configured to:
publish the respective secret share to a respective database system of a plurality of database systems such that no single entity has access to multiple data secret shares and multiple MAC secret shares in one database system.

20. The system of claim 14, wherein the hardware processor is further configured to compute MAC_{v2} by:

computing $V \times R \times MAC_{key2}$ to determine $MAC_{v2} \bmod M$, wherein V is the data, MAC_{key2} is the message authentication code key known by the plurality of SMPC compute nodes, R is the random number within the public mod value (M), and R^{-1} is the inverse of R in mod M .

21. A non-transitory computer readable medium storing thereon computer executable instructions for preserving data integrity when integrating secure multiparty computation (SMPC) and blockchain technology, including instructions for:

- splitting, via a data publisher, data into a plurality of data secret shares using a SMPC protocol;

- generating, using the SMPC protocol, a plurality of message authentication code (MAC) secret shares of MAC condition parameters that ensure integrity of the data by:

 - generating, via the data publisher, a random number within a public mod value known by a plurality of SMPC compute nodes;

 - computing a message authentication code (MAC_{v2}) based on the data, the random number, and a message authentication code key known by the plurality of SMPC compute nodes;

 - determining an inverse of the random number; and

 - including additional secret shares of the inverse and the MAC_{v2} in the plurality of MAC secret shares for encryption and publication on the blockchain, wherein each respective secret share of the inverse and the MAC_{v2} is matched with a corresponding data secret share in the plurality of data secret shares;

- wherein each data secret share of the plurality of data secret shares and each MAC secret share of the plurality of MAC secret shares is assigned to an SMPC compute node of the plurality of SMPC compute nodes; and

- for each respective data secret share and MAC secret share:

 - identifying, from the plurality of SMPC compute nodes, a respective SMPC compute node assigned to the respective data secret share and MAC secret share;

 - retrieving a respective public key of the respective SMPC compute node;

 - encrypting the respective data secret share and MAC secret share using the respective public key; and

 - publishing the encrypted respective data secret share and MAC secret share on a blockchain.