CLAIMS

1. A computer implemented system for a trusted execution environment maintaining an isolated data processing subsystem, the system comprising:

    a computer readable memory having a protected memory region that is encrypted such that it is inaccessible to both an operating system and kernel system, the protected memory region including at least a data storage region and a data processing subsystem storage region maintaining the isolated data processing subsystem;

    a computer readable cache memory; and

    a secure enclave data processor configured to provide:

        a partner data receiver configured to separately receive one or more data sets from a plurality of corresponding partner computing devices, each of the one or more data sets digitally signed by a private key corresponding to each of the partner computing devices, the partner data receiver configured to validate the digital signature of the partner computing device;

        the partner data receiver configured to: (i) load a portion of the protected memory region into the computer readable cache memory, (ii) record the one or more validated data sets into the loaded portion of the data storage region in the protected memory region, and to (iii) unload and encrypt the portion of the protected memory region subsequent to the recordal; and

        a data processing subsystem interaction engine configured to transmit a query data message to the data processing subsystem and to receive an output data structure generated by the data processing subsystem generated based on the one or more data sets stored in the protected memory region.

2. The system of claim 1, wherein the isolated data processing subsystem maintains an isolated machine learning data model architecture comprising a series of interconnected computing nodes that operate in concert to generate the output data structure responsive to the query data message, the interconnected computing nodes including a series of weights and filters representing a dynamically modified function that is trained over a number of training epochs using at least a portion of the one or more validated data sets in the data storage region in the protected memory region as training sets or validation sets representing data from at least two partner computing devices of the plurality of partner computing devices, the query data message triggering the dynamically modified function to generate the output data structure.

3. The system of claim 2, wherein the secure enclave data processor is configured to provide:

   a partition controller engine configured to provision one or more secure enclave sub processors and to transmit to each of the one or more secure enclave sub processors a partition of the protected memory region;

   the one or more secure enclave sub processors configured to process the corresponding partition of the protected memory region using a local copy of the machine learning data model architecture to generate one or more parameter update data structures; and

   a partition aggregation engine configured to receive, from each of the one or more secure enclave sub processors, the one or more parameter update data structures, and to process the one or more parameter update data structures to refine at least one parameter of the machine learning data model architecture, the machine learning data model architecture for distribution to the one or more secure enclave sub processors to update the corresponding local copy of the machine learning data model architecture.

4. The system of claim 3, wherein there are at least two secure enclave sub processors, including a first secure enclave sub processor and a second secure enclave sub processors that are configured to transmit one or more parameter update data structures directly between one another to update the corresponding local copy of the machine learning data model architecture.

5. The system of claim 1, wherein the partner data receiver is further configured to generate public/private key pairs each corresponding to a partner computing device of the one or more corresponding partner computing devices, and to transmit the private key corresponding to corresponding to a partner computing device to the partner computing device.

6. The system of claim 1, wherein the computer readable memory having the protected memory region is stored on DRAM.

7. The system of claim 1, wherein the key required to decrypt the protected memory region into the computer readable cache memory is stored within the secure enclave data processor and not accessible outside the secure enclave data processor.

8. The system of claim 1, wherein the key required to decrypt the protected memory region into the computer readable cache memory is originally generated with a nonce term, and the nonce term is stored within the secure enclave data processor and not accessible outside the secure enclave data processor.

9. The system of claim 1, wherein a remote attestation process is periodically conducted by a secure enclave data processor to validate security of the system, the remote attestation process includes transmitting a remote attestation payload to the secure enclave data processor that includes a Diffie Hellman message.

10. The system of claim 1, wherein a remote attestation process is periodically conducted by a secure enclave data processor to validate security of the system, the remote attestation process includes the secure enclave data processor generating a remote attestation transcript data structure and transmitting the remote attestation transcript data structure along with a signed challenge payload and a new Diffie Hellman message payload.

11. A computer implemented method for a trusted execution environment maintaining an isolated data processing subsystem, the method operating on a computer readable memory having a protected memory region that is encrypted such that it is inaccessible to an operating system and kernel system of a computing device implementing the trusted execution environment, the protected memory region including at least a data storage region and a data processing subsystem storage region maintaining the isolated data processing subsystem, the method comprising:

    receiving one or more data sets from one or more corresponding partner computing devices, each of the one or more data sets digitally signed by a private key corresponding to each of the partner computing devices;

    validating the digital signature of the partner computing device;

    loading a portion of the protected memory region into a computer readable cache memory;

    recording the one or more validated data sets into the loaded portion of the data storage region in the protected memory region;

    unloading and encrypting the portion of the protected memory region subsequent to the recordal;

    transmitting a query data message to the data processing subsystem; and

    processing the query data message based on a processing function that receives as an input the one or more data sets from the one or more corresponding partner computing devices or an aggregate thereof to generate an output data structure.

12. The method of claim 11, wherein the isolated data processing subsystem maintains a isolated machine learning data model architecture comprising a series of interconnected computing nodes that operate in

concert to generate the output data structure responsive to the query data message, the interconnected computing nodes including a series of weights and filters representing a dynamically modified function that is trained over a number of training epochs using at least a portion of the one or more validated data sets into data storage region in the protected memory region as training sets or validation sets; and wherein

the processing of the query data message based on a processing function includes applying the dynamically modified function to the query data message to generate the output data structure.

13. The method of claim 12, comprising provisioning one or more secure enclave sub processors and transmitting to each of the one or more secure enclave sub processors a partition of the protected memory region;

processing the corresponding partition of the protected memory region using a local copy of the machine learning data model architecture to generate one or more parameter update data structures;

receiving, from each of the one or more secure enclave sub processors, the one or more parameter update data structures;

processing the one or more parameter update data structures to refine at least one parameter of the machine learning data model architecture; and

distributing refined at least one parameter to the one or more secure enclave sub processors to update the corresponding local copy of the machine learning data model architecture.

14. The method of claim 11, wherein the partner data receiver is further configured to generate public/private key pairs each corresponding to a partner computing device of the one or more corresponding partner computing devices, and the method comprises:

transmitting the private key corresponding to corresponding to a partner computing device to the partner computing device.

15. The method of claim 11, wherein the computer readable memory having the protected memory region is stored on DRAM.

16. The method of claim 11, wherein the key required to decrypt the protected memory region into the computer readable cache memory is stored within a secure enclave data processor and not accessible outside the secure enclave data processor.

17. The method of claim 11, wherein the key required to decrypt the protected memory region into the computer readable cache memory is originally generated with a nonce term, and the nonce term is stored within the secure enclave data processor and the nonce term is not accessible outside the secure enclave data processor.

18. The method of claim 11, comprising periodically conducting a remote attestation process for security validation, the remote attestation process including transmitting a remote attestation payload to the secure enclave data processor that includes a Diffie Hellman message.

19. The method of claim 11, comprising periodically conducting a remote attestation process for security validation, the remote attestation process including generating, by the secure enclave data processor, a remote attestation transcript data structure and transmitting the remote attestation transcript data structure along with a signed challenge payload and a new Diffie Hellman message payload to the one or more computing devices.

20. A non-transitory computer readable medium, storing machine interpretable instructions which when executed by a processor, cause the processor to perform a computer implemented method for a trusted execution environment maintaining an isolated data processing subsystem, the method operating on a computer readable memory having a protected memory region that is encrypted such that it is inaccessible to an operating system and kernel system of a computing device implementing the trusted execution environment, the protected memory region including at least a data storage region and a data processing subsystem storage region maintaining the isolated data processing subsystem, the method comprising:

    receiving one or more data sets from one or more corresponding partner computing devices, each of the one or more data sets digitally signed by a private key corresponding to each of the partner computing devices;

    validating the digital signature of the partner computing device;

    loading a portion of the protected memory region into a computer readable cache memory;

    recording the one or more validated data sets into the loaded portion of the data storage region in the protected memory region;

    unloading and encrypting the portion of the protected memory region subsequent to the recordal;

    transmitting a query data message to the data processing subsystem; and

processing the query data message based on a processing function that receives as an input the one or more data sets from the one or more corresponding partner computing devices or an aggregate thereof to generate an output data structure.