CLAIMS

1. A system for periodic management of cryptographic keys, the system comprising:

a plurality of computer processors, comprising a first computer processor in a first data-stream publisher device, a second computer processor in a data-stream back-end device, and a third computer processor in a first subscriber device;

a first memory communicatively coupled with the first computer processor, a second memory communicatively coupled with the second computer processor, and a third memory communicatively coupled with the third computer processor, the second memory storing computer-executable instructions that, when executed, cause at least:

the second computer processor to receive a first input associating a first set of subscribers with a first data stream published by the first data-stream publisher device;

the second computer processor to receive, via a key-proxy service, a first cryptographic key;

the second computer processor to transmit a first confirmation, indicating that the first cryptographic key is ready for use;

the second computer processor to release, via the key-proxy service, the first cryptographic key to at least the first set of subscribers comprising the first subscriber device;

the second computer processor to receive a second input, from a publishing user, associating a second set of subscribers with the first data stream published by the first data-stream publisher device, wherein the second set of subscribers excludes the first subscriber device;

the second computer processor to receive, via the key-proxy service, a second cryptographic key after a predetermined amount of time has elapsed, wherein the second cryptographic key is ineffective to decrypt data that can be decrypted using the first cryptographic key;

the second computer processor to transmit a second confirmation, indicating that the second cryptographic key is ready for use; and

the second computer processor to release the second cryptographic key to at least the second set of subscribers, whereby the first subscriber device, included in the first set of subscribers but excluded from the second set of subscribers, is unable to decrypt, using the first cryptographic key, encrypted data that requires the second

cryptographic key to decrypt, irrespective of access to the first data stream by the first subscriber device.

2. The system of claim 1, wherein the first computer processor is further configured to emit a signal to instruct at least one device, comprising the first data-stream publisher device, to cease using a first key-generation algorithm.

3. The system of claim 1, wherein, in response to the second input, the second computer processor is further configured to replace the first cryptographic key with the second cryptographic key.

4. The system of claim 1, wherein the second input comprises:

selection, by the publishing user, of the first subscriber device, from the first set of subscribers, and

revocation of an access permission previously granted to the first subscriber device, such that the second set of subscribers excludes the first subscriber device.

5. The system of claim 1, wherein the second computer processor is further configured to transmit a third confirmation that at least one of the first cryptographic key or the second cryptographic key is available, upon having been loaded by the key-proxy service.

6. The system of claim 1, wherein the first data stream published by the first data-stream publisher device is associated with a smart contract on a decentralized, permissionless blockchain platform associated with the key-proxy service, and wherein the first data stream is tokenized via the smart contract.

7. The system of claim 6, wherein to release the first cryptographic key, the second computer processor is further configured to associate the first cryptographic key with the smart contract; and wherein to release the second cryptographic key, the second computer processor is further configured to replace the first cryptographic key with

the second cryptographic key as a current cryptographic key associated with the smart contract.

8. The system of claim 7, wherein the second computer processor is configured to receive, via the key-proxy service, a request from the third computer processor for the current cryptographic key associated with the smart contract.

9. The system of claim 8, wherein the first data stream published by the first data-stream publisher device is associated with a smart contract on a blockchain platform associated with the key-proxy service, and wherein the first data stream is tokenized via the smart contract.

10. The system of claim 7, wherein to release the first cryptographic key, the second computer processor is further configured to associate the first set of subscribers with the smart contract; and wherein to release the second cryptographic key, the second computer processor is further configured to associate the second set of subscribers with the smart contract, in lieu of the first set of subscribers.

11. The system of claim 1, wherein the first cryptographic key and the second cryptographic key are generated by the first data-stream publisher device including the first computer processor and are received by the second computer processor from the first data-stream publisher device.

12. The system of claim 1, wherein the first computer processor is further configured to transmit an announcement to at least one subscriber device, comprising the third computer processor, that a new indexed cryptographic key is available.

13. The system of claim 12, wherein the announcement is transmitted in response to a request from the at least one subscriber device to the first computer processor.

14. The system of claim 1, wherein the first cryptographic key and the second cryptographic key are received from a cryptographic key generator in the system

using at least one entropy source, wherein the first cryptographic key is transmitted via the key-proxy service to the second computer processor by the first data-stream publisher device as the first confirmation indicating that the first cryptographic key is ready for use, and wherein the second cryptographic key is transmitted via the key-proxy service to the second computer processor by the first data-stream publisher device as the second confirmation indicating that the second cryptographic key is ready for use.

15. An apparatus for periodic management of cryptographic keys at a data-stream publisher, the apparatus comprising:

at least one memory device configured to store computer-executable instructions; and

at least one computer processor communicatively coupled with the at least one memory, the at least one computer processor capable of being configured, via at least the computer-executable instructions, to perform operations comprising:

load a first cryptographic key;

transmit the first cryptographic key;

receive a first confirmation that the first cryptographic key is ready for use;

encrypt at least one first data payload, wherein the at least one first data payload, once encrypted, requires the first cryptographic key for any subsequent decryption of the at least one first data payload;

publish, via a data stream, the at least one first data payload as encrypted;

load a second cryptographic key, responsive to at least a first factor comprising an indication that a predetermined amount of time has elapsed;

transmit the second cryptographic key;

emit a first notification of whether the second cryptographic key is ready for use;

receive a second confirmation that the second cryptographic key is ready for use;

encrypt at least one second data payload, wherein the at least one second data payload, once encrypted, requires the second cryptographic key

for any subsequent decryption of the at least one second data payload, and wherein the first cryptographic key is ineffective to decrypt the at least one second data payload; and

publish, via the data stream, the at least one second data payload as encrypted.

16. The apparatus of claim 15, wherein the at least one computer processor is further configured to load the second cryptographic key, in response to a second factor comprising a replacement of the first cryptographic key with the second cryptographic key.

17. The apparatus of claim 15, wherein to load the first cryptographic key, the at least one computer processor is further configured to perform a first key-generation algorithm.

18. The apparatus of claim 17, wherein to load the second cryptographic key, the at least one computer processor is further configured to perform a second key-generation algorithm, in response to a signal to cease using the first key-generation algorithm.

19. The apparatus of claim 17, wherein to perform the first key-generation algorithm, the at least one computer processor is further configured to source entropy from at least one hardware source on-board the apparatus.

20. The apparatus of claim 17, wherein to perform the first key-generation algorithm, the at least one computer processor is further configured to source entropy from at least one hardware source remote from the apparatus and communicatively coupled with the at least one computer processor via a network interface, wherein the at least one hardware source comprises another apparatus.

21. The apparatus of claim 15, wherein to load the first cryptographic key, the at least one computer processor is further configured to receive the first cryptographic key.

22. The apparatus of claim 15, wherein to encrypt the at least one first data payload, the at least one processor is configured to perform a first encryption algorithm for publication via the data stream, wherein, before the first encryption algorithm is performed, the at least one first data payload has been encrypted using one of the first encryption algorithm or a second encryption algorithm, and wherein a third cryptographic key is required for decryption of the encrypted data payload.

23. The apparatus of claim 15, wherein the at least one computer processor is further configured to announce at least one of the first cryptographic key or the second cryptographic key as being available.

24. The apparatus of claim 15, wherein the data stream is associated with a smart contract on a blockchain platform associated with a key-proxy service, and wherein the data stream is tokenized via the smart contract.

25. The apparatus of claim 15, further comprising an environmental sensor, a biotelemetry monitor, a geolocation device, a position-tracking device, an inventory-accounting device, a traffic-counter device, a flow-measurement device, or a combination thereof.

26. The apparatus of claim 15, the operations further comprising:

associating the first cryptographic key with a first index value;

associating the second cryptographic key with a second index value; and

transmitting, with the second cryptographic key, the second index value to a back-end service comprising a key-proxy service.

27. The apparatus of claim 26, the operations further comprising:

emit a second notification, to the back-end service, that the first cryptographic key is ready for use, in response to a first request of whether the first cryptographic key is ready for use; and

emit a third notification, to the back-end service, that the first cryptographic key is ready for use, in response to a second request of whether the first cryptographic key is ready for use.

28. An apparatus for periodic management of cryptographic keys at a back-end service provider, the apparatus comprising:

at least one memory device configured to store computer-executable instructions; and

at least one computer processor communicatively coupled with the at least one memory, the at least one computer processor capable of being configured, via at least the computer-executable instructions, to perform operations comprising:

receive a first input associating a first set of subscribers with a first data stream published by a data-stream publisher device;

receive, via a key-proxy service, a first cryptographic key;

transmit a first confirmation, indicating that the first cryptographic key is ready for use;

release, via the key-proxy service, the first cryptographic key to at least the first set of subscribers comprising a first subscriber device;

receive a second input, from a publishing user, associating a second set of subscribers with the first data stream published by the data-stream publisher device, wherein the second set of subscribers excludes the first subscriber device;

receive, via the key-proxy service, a second cryptographic key after a predetermined amount of time has elapsed, wherein the second cryptographic key is ineffective to decrypt data that can be decrypted using the first cryptographic key;

transmit a second confirmation, indicating that the second cryptographic key is ready for use; and

release the second cryptographic key to at least the second set of subscribers.

29. A non-transitory computer-readable storage device having instructions stored thereon, wherein the instructions, when executed by at least one computer processor, cause the at least one computer processor to perform operations comprising:

load a first cryptographic key;

transmit the first cryptographic key to a back-end service;

confirm that the first cryptographic key is ready for use;

encrypt at least one first data payload, wherein the at least one first data payload, once encrypted, requires the first cryptographic key for any subsequent decryption of the at least one first data payload;

publish, via a data stream, the at least one first data payload as encrypted;

load a second cryptographic key, responsive to at least a first factor comprising an indication that a predetermined amount of time has elapsed;

transmit the second cryptographic key to the back-end service;

confirm that the second cryptographic key is ready for use;

encrypt at least one second data payload, wherein the at least one second data payload, once encrypted, requires the second cryptographic key for any subsequent decryption of the at least one second data payload, and wherein the first cryptographic key is ineffective to decrypt the at least one second data payload; and

publish, via the data stream, the at least one second data payload as encrypted.