CLAIMS

1. A method for lightweight encryption, comprising:

initializing a first plurality of buffers on an encrypting device and a decrypting device;

receiving a message for encrypting;

incrementing a first counter;

determining a message byte (n), wherein n is equal to the first counter incremented;

determining a first offset buffer value corresponding to a location indicated by the first counter;

determining a first dynamic key buffer location by summing the message byte (n) and the first offset buffer value;

resetting a dynamic key buffer index; and

ending the method when a last message byte is received.

2. The method of claim 1, wherein the received message is encrypted, by the encrypting device, by utilizing a shared key that is repeatedly updated using transmitted data as a randomizing component in generating the shared key, wherein the shared key is a product of a transmission history between two points, and wherein the method further comprises transmitting the encrypted message, and decrypting the encrypted message on the decrypting device utilizing the shared key.

3. The method of claim 1, wherein before the resetting step the method comprises determining an encrypted message value corresponding with a first dynamic key buffer value at the first dynamic key buffer location.

4. The method of claim 3, wherein the first plurality of buffers comprises:

a first offset buffer;

a first dynamic key buffer; and

the first dynamic key buffer index.

5. The method of claim 4, wherein, before the resetting of the dynamic key buffer index, the first dynamic key buffer is shuffled, and wherein the method, after the resetting, further comprises continuing incrementing the first counter until the last message byte is received.

6. The method of claim 5, wherein the first dynamic key buffer is shuffled by:

swapping a last value in the first dynamic key buffer with the encrypted message value; and

shuffling the first dynamic key buffer upward to replace an encrypted value location.

7. The method of claim 5, wherein resetting the first dynamic key buffer index generates a location index of each value from 0 to 255 in the first dynamic key buffer after the shuffling.

8. The method of claim 4, wherein the initializing utilizes an identical seed value on the encrypting device and the decrypting device.

9. The method of claim 8, wherein the first offset buffer, the first dynamic key buffer, and the first dynamic key buffer index are initialized with 256 values from 0 to 255 that are randomly shuffled using the identical seed value.

10. The method of claim 2, wherein the decrypting the encrypted message comprises:

initializing a second plurality of buffers on the decrypting device;

receiving the encrypted message;

incrementing a second counter for (m);

determining an encrypted message byte (m);

determining a second dynamic key buffer location indicated by the encrypted message byte (m);

determining a value corresponding with the second dynamic key buffer location;

determining a second offset buffer value for (m); and

determining a decrypted message value by subtracting the second offset buffer value for (m) from the value corresponding with the second dynamic key buffer location.

11. The method of claim 10, wherein the second plurality of buffers comprises:

a second offset buffer;

a second dynamic key buffer; and

a second dynamic key buffer index.

12. The method of claim 11, further comprising:

shuffling the second dynamic key buffer;

resetting the second dynamic key buffer index; and

continuing to the incrementing the second counter until a last encrypted message byte is received.

13. The method of claim 12, wherein the shuffling the second dynamic key buffer comprises:

swapping a last value in the second dynamic key buffer with the encrypted message value; and

shuffling the second dynamic key buffer upward to replace an encrypted value location.

14. The method of claim 12, wherein resetting the second dynamic key buffer index generates a location index of each value from 0 to 255 in the second dynamic key buffer after the shuffling.