Abstract—Wireless Sensor Network (WSN) consists of a large number of small sensor nodes (SNs) that are usually deployed in hostile environments. These nodes are vulnerable to a number of security attacks and it is difficult to implement complex cryptographic schemes in SNs due to their resource constrained nature. Hence, WSNs need energy efficient and secure communication schemes in terms of storage space, power consumption and operation speed. In this work, we propose a secure and lightweight encryption scheme based on chaotic map and genetic operations. We also present some initial results of security strength and performance evaluation experiments.

I. INTRODUCTION

The application domains of WSNs have gradually increased with the continuous development of powerful wireless technologies, intelligent sensors and micro-electronic-mechanical systems (MEMS). Now-a-days, WSNs intended for various industrial, medical and military applications keep challenging the researchers in the design of secure architectures and protocols. Specially, the use of sensors in critical systems such as nuclear power plants, aircrafts and healthcare requires effective mechanisms to ensure the authenticity, confidentiality and integrity of the transmitted data [1]. However, security is a challenging issue in WSNs since sensor networks usually deployed in harsh environments. Moreover, small memories, weak processors, limited communication range of sensor nodes introduce a number of problems in implementing traditional cryptographic schemes in sensor networks. Therefore, WSNs require efficient encryption schemes that can maintain the trade-off among security, performance and cost.

The encryption algorithms in WSNs are classified into three categories: compact hardware oriented cryptosystems (e.g. KATAN and KATANTAN), conventional block ciphers (e.g. XXTEA and SkipJack) and lightweight block ciphers (e.g. LED and TWINE). Among these security protocols, the compact hardware oriented cryptographic schemes are not suitable for WSNs due to high energy consumption and memory usage [2]. Again, most of the traditional block ciphers adopted in WSNs are vulnerable to a number of security threats [3]. Therefore, the current researches focus on lightweight block ciphers to overcome these problems. However, a major limitation of security protocols in WSNs is that most of them can not be used for both text and image data encryption. This is because image data is different from text due to some intrinsic features such as strong correlation between adjacent pixels and high redundancy. Hence, the encryption scheme should be robust, fast and computationally secure. In this work, we propose a chaotic map and genetic operation based lightweight encryption scheme for tiny sensor devices which is suitable for both text and image data encryption.

II. THE PROPOSED ENCRYPTION SCHEME

A. Key Establishment Phase

We assume that the elliptic curve and chaotic map parameters \( (p, G(x,y), A, B, m, N, \mu \) and \( \beta \) are predistributed securely among the SNs in the network. Now, using point addition and point doubling elliptic curve operations, the SNs can generate a list of elliptic curve points known as key pool [4]. When a node requires to send data packets to another node, it randomly picks a key from the list and generates the hash code of the key using a hash function. This code is then sent to the destination node. The destination node retrieves the key by matching the received code with the hash codes generated for each point of its key pool. Upon successful retrieval of the secret key, it acknowledges the source node with a reply message.

B. Generation of Pseudorandom Bit Sequence

We use N-logistic tent map to generate the pseudorandom bit sequence in our proposed scheme [5]. One of the main benefits of this chaotic function is that it supports a wider data range with lower digital computational cost. The N-logistic tent map is defined by the following equations:

\[
\begin{align*}
    x_{n+1} &= \mu x_n(N - x_n/m)/N - y_n/2 \\
    y_{n+1} &= \beta(N - |N - y_n|)
\end{align*}
\]

where, \( x \in (0, 2^N), \mu \in [0, 4], y \in (0, 2\times N), \beta \in [1, 2], N = 2^k \) and \( m = 2^k \) with integers \( K \) and \( k \). The seed key is the set \( \{ x_1, y_1, \mu, \beta, m, N \} \), where, initial \( x_1 \) and \( y_1 \) are obtained in the previous phase.

C. The Encryption Procedure

At first, we divide the random bit sequence into 256-bits blocks and each block is subdivided into two 128-bits blocks. Then we calculate the number of 1’s in each byte as well as the sum of 1’s for each two consecutive bytes in the second half of the random bit sequence. After that, we convert the plaintext (or plain image) into their corresponding binary codes and group them into blocks of 128-bits. This binary code is xor-ed with the first block of the random bit sequence since the additive cipher XOR is secure for same length of key-stream. Then, the first genetic operation i.e., mutation is performed on each byte of xor-ed plaintext. For example, if the number of 1’s in the first byte of the second half of random bit sequence is 7, we mutate the 7th and 8th number bits in the first byte of the xor-ed plaintext. The mutated plaintext is further crossovered as shown in the Fig. 1 to generate the ciphertext. This crossover operation is done repeatedly (e.g., \( B_1-B_2-B_3-B_4, B_2-B_3-B_4-B_5 \ldots B_{n-2}-B_{n-1}-B_n-B_1 \)) so that each of the byte performs the operation for twice in order to add relatively fair diversity. The decryption process is simply reverse of the encryption procedure.
III. SECURITY AND PERFORMANCE ANALYSIS

We have tested our proposed scheme against a number of security attacks. Here, we present some experimental outcomes of security and performance analysis. We have used a gray-level image of size $128 \times 128$ in our experiments.

A. Key-space and Statistical Analysis

In our proposed encryption scheme, the secret keys set is $\{ x_i, y_i, \mu, \beta, m, N \}$, where $\mu$ and $\beta$ are double precision numbers while $x, y, N$ and $m$ are integer parameters having the following range: $\{ x, y, N \} \in [1, 2^{128}], m \in [1, 2^{64}]$. According to IEEE floating-point standard, the computational precision of the 64-bit double precision number is about $10^{-15}$. Therefore, the complete key-space of the proposed encryption scheme is $7.268 \times 10^{164} \approx 2^{548}$.

We have performed histogram analysis and correlation of adjacent pixels analysis on our proposed encryption scheme. The Fig. 2 shows the histograms of both plain image and cipher image. From the figure, we see that the information in cipher image is fairly uniform and significantly different from that of the plain image and hence can resist statistical attack well. To test the correlation of adjacent pixels, we have calculated the correlation of two horizontally, two vertically and two diagonally adjacent pixels in the cipher image. The correlations of two horizontally, vertically and diagonally adjacent pixels are 0.0027, 0.0019 and 0.007 respectively for the cipher image while the values are 0.896, 0.947 and 0.864 respectively for the plain image. The results indicate that the correlation between two adjacent pixels in the cipher image is negligible.

B. Performance Analysis

We have also implemented our encryption scheme in Mica2 sensor mote and compared the results with LED and TWINE protocols. The results in the Table I show that our proposed encryption scheme performs better than LED and TWINE in terms of CPU elapsed time (8.547ms) using only 62396 CPU cycles. The elapsed time and CPU cycles are almost double for TWINE and nine times higher for LED protocol with respect to our proposed protocol. In case of total memory consumption, our proposed scheme uses slightly more memory (204 bytes) compared to TWINE but it is less than that of LED.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>CPU Cycles</th>
<th>Time (ms)</th>
<th>RAM (bytes)</th>
<th>ROM (bytes)</th>
<th>Total Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>LED</td>
<td>589652</td>
<td>78.972</td>
<td>378</td>
<td>5970</td>
<td>6348</td>
</tr>
<tr>
<td>TWINE</td>
<td>128996</td>
<td>17.477</td>
<td>384</td>
<td>5280</td>
<td>5664</td>
</tr>
<tr>
<td>Proposed</td>
<td>62396</td>
<td>8.547</td>
<td>542</td>
<td>5326</td>
<td>5868</td>
</tr>
</tbody>
</table>

IV. CONCLUSION AND FUTURE WORKS

This study presents a lightweight encryption scheme based on chaotic map and genetic operations. Our proposed scheme is secure, lightweight and suitable for use in WSNs. In our future work, we will implement the proposed encryption scheme for audio and video data encryption. Moreover, we will implement the protocol in large scale sensor networks to evaluate the overall throughput, latency and communication cost.

ACKNOWLEDGMENT

I would like to acknowledge the contributions of my PhD supervisor Dr. Vallipuram Muthukkumarasamy to this work.

REFERENCES