IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 27, 2020, accepted November 3, 2020, date of publication November 6, 2020, date of current version November 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3036383

Reversible Multiple Items Authentication Scheme for WSNs

GUANGYONG GAO[1,2], BIN WU[2], YUXIANG WANG[1], ZHIQIANG ZHAO[2], LIYA XU[2], AND CHAO YIN[2]

[1]School of Computer and Software, Nanjing University of Information Science & Technology, Nanjing 210044, China
[2]School of Information Science & Technology, Jiujiang University, Jiujiang 332005, China

Corresponding author: Chao Yin (cs_yinchao@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61662039 and Grant 61662038, in part by the Jiangxi Key Natural Science Foundation under Grant 20192ACBL20031, in part by the Startup Foundation for Introducing Talent of Nanjing University of Information Science & Technology (NUIST) under Grant 2019r070, and in part by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) Fund.

ABSTRACT In this article, a reversible authentication scheme for wireless sensor networks (WSNs) is proposed aimed at multiple items authentication and good imperception of transmitted data. Firstly, the WSNs data stream is divided into some authentication groups, and each authentication group is composed of a generator group and a carrier group. Then, the cyclical redundancy check (CRC) code of the generator group is generated as the authentication information, which is composed of the CRC codes of all collected items in the generator group. In the carrier group, first, to decrease the visual perception introduced by information embedding, the beginning parameter T_m and the end parameter T_p are used to select the fluctuation region of the prediction-error histogram (PEH) where large prediction errors occur. Then, using the prediction-error-based histogram shifting method, the authentication information is reversibly embedded into the fluctuation region. Experimental results and analysis demonstrate that, compared with up-to-date homogeneous schemes, the proposed scheme achieved better performance on invisibility, false positive rate, type of authenticated items, and attraction to attackers.

INDEX TERMS Wireless sensor networks (WSNs), multiple items authentication, fluctuation region, prediction-error histogram (PEH).

I. INTRODUCTION

A core issue of wireless sensor networks (WSNs) [1], [2] is data integrity authentication. Traditionally, encryption technology is applied to maintain the security of WSNs [3]–[8]. Karlof *et al.* [3] developed the first security protocol for the link layer of WSNs called TinySec, which assures data confidentiality and provides data integrity authentication. In another study [4], Ullah *et al.* proposed a lightweight certificateless signcryption approach for crowdsourced IIoT applications to enhance security and decrease the computational cost and communication overhead. The security and efficiency of the proposed approach are based on a hyperelliptic curve cryptosystem. Seo *et al.* [5] proposed a certificateless-effective key management (CL-EKM) protocol for secure communication in dynamic WSNs characterized by node mobility. The CL-EKM supports efficient key updates when a node leaves or joins a cluster and ensures forward and backward key secrecy. Chen *et al.* [6] proposed a recoverable concealed data aggregation scheme based on privacy homomorphism encryption for data integrity in WSNs, which provides better security compared with traditional aggregation and reduced transmission overhead. However, applying encryption technology to WSNs does not work because of the limitation of sensor node resources.

Later, the data hiding technology applied to image security was used for the data authentication of WSNs [10]–[14]. Compared with encryption technology, data hiding is more lightweight on resource consumption. A data hiding scheme was proposed in one study [10]. This scheme utilizes the property of micro-errors of sensor data to embed information, and, so long as the introduced error is within a limited scope, the WSNs data can be used normally. Chen and Hou [11] proposed a compressed sensing (CS)-based watermark

The associate editor coordinating the review of this manuscript and approving it for publication was Mahdi Zareei.

cryptosystem for WSNs, leveraging the characteristic that CS reconstruction is sensitive to measurement noise. In the front-end sensor, a low-dimension watermark was embedded in measurements. In the back-end solver, the scheme uses a CS-based watermark decryption/reconstruction engine for the Internet of Things (IoT) gateway. In another study [12], a fragile chain-watermarking scheme was developed. This scheme regards WSN data as a stream chained by many groups, and watermarking is generated by the repeated use of the hash function for different groups. The experimental results indicated that the fragile chain-watermarking scheme can verify the integrity of WSN data. It should be noted that the above data hiding schemes are not reversible.

In some special applications, such as medical and military fields, the authenticated data is not allowed to be modified because reversible data hiding (RDH) technology can restore the original data thoroughly after the embedded information is extracted [15], [16]. Therefore, for these special fields, RDH technology is very suitable [17], [18]. Currently, RDH technology is also applied to secure communication for WSN data, and some studies have been published [19]–[23]. Shi and Xiao [19] proposed a reversible authentication scheme based on group strategy. Each authentication group is determined dynamically by synchronization points, and the data stream can be totally recovered after the authentication information composed of the hash value of the authentication group is extracted. Wang *et al.* [20] aimed at designing a data authentication mechanism integrated with an energy-conserving routing scheme for WSNs to protect the integrity of receiving sensed data while at the same time conserving the battery energy of the sensor nodes. An energy-conserving reversible and irreversible digital watermarking hybrid scheme was proposed to balance the energy consumption and the robustness of the security. In another study [21], a reversible authentication scheme was achieved based on the cyclical redundancy check (CRC) and odd-even invariability, but this scheme has a high false positive rate (FPR). To solve the contradiction between high energy consumption and the constrained resources of the sensor network, Jiang *et al.* [22] proposed a recoverable data fusion protocol that ensures data integrity and confidentiality based on reversible digital watermarking and homomorphic encryption technology. In another study [23], for the secure transmission of wireless body area networks, Liu *et al.* proposed a lightweight integrity authentication scheme based reversible watermark. The shortages of this scheme are that it statically divides the data stream into different data packets, and the total authentication group is buffered by the sensor node to achieve the watermark generation and embedding.

In summary, in the existing literature, multiple item authentication and the imperception of the transmitted data have not been considered. To address the two issues, a novel reversible multiple item authentication scheme for WSNs is proposed in this article. The multiple watermarks are generated by calculating the CRC codes of multiple items of the generator group. In the fluctuation region of the prediction-error histogram (PEH) of the carrier group, the watermarks are reversibly embedded by applying the PEH shifting algorithm. Experimental results and analysis demonstrate that the proposed scheme achieved better performance than up-to-date reversible authentication schemes for the secure transmission of WSNs data.

The main contributions of this article can be summarized as follows:

1) Different from the data integrity authentication of only one category of item in traditional WSNs schemes, the data integrity authentications of multiple categories of items are conducted.
2) The authentication information is embedded only into the fluctuation region of the PEH of the carrier group, which ensures good imperceptibility of the transmitted data.

The rest of this article is organized as follows: Section II describes the proposed algorithm. Experimental results are given in Section III, and Section IV presents the conclusion and summarizes this article.

II. PROPOSED ALGORITHM

In this article, we propose a new reversible multiple items authentication scheme for WSNs. As shown in Figure 1, the proposed method contains two phases: 1) Group division and watermark embedding, which are conducted by the sensor node; and 2) tampering detection and data restoration, which are done by the convergence node. In the first phase, the WSNs data stream is first divided into different authentication groups, each of which is composed of a generator group and a carrier group. Then, the CRC code, as a watermark, which is calculated by the segments of multiple items in the generator group, is embedded into the carrier group by the PEH shifting method. In the second phase, according to the comparison result between the watermark extracted from the carrier group and that generated by the generator group, the decoder decides whether to restore the original data or whether tampering is detected.

A. SYSTEM MODEL

A common WSNs system model is composed of a sensor node, a transmission node, and a convergence node, which are shown in Figure 2. The sensor nodes are responsible for periodically collecting environmental data of the detected region, such as temperature, pressure, humidity, and illumination, and the watermark used is embedded as authentication information. The embedded WSNs data are transmitted to the convergence node by the transmission nodes. At the convergence node end, the works to be conducted involve watermark extraction, data tampering detection, and data recovery.

B. AUTHENTICATION GROUP

In the proposed scheme, the WSNs data stream collected by the sensor node is denoted by T, which is divided into a series of authentication groups. An example of an authentication

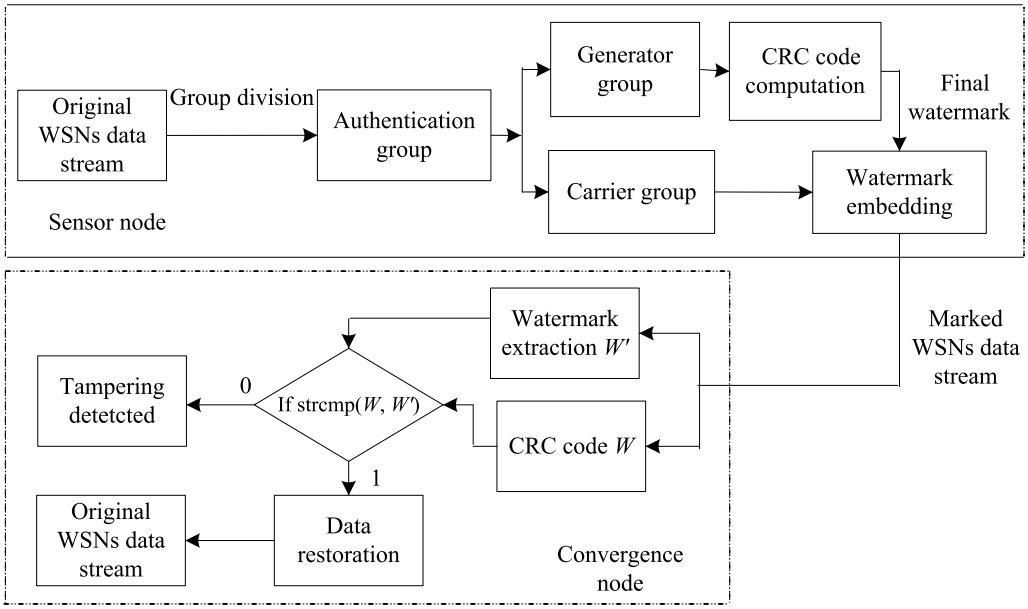

FIGURE 1. The framework of the proposed method.

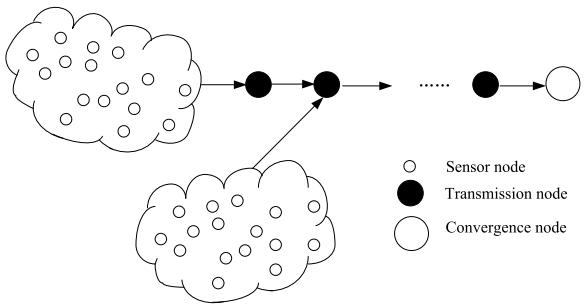

FIGURE 2. A simple WSNs system model.

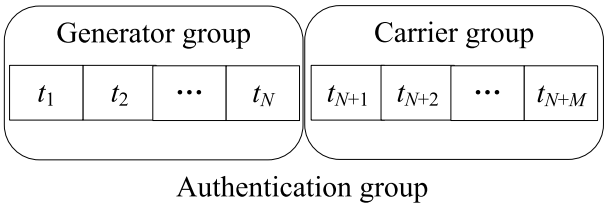

FIGURE 3. An example of an authentication group.

group is presented in Figure 3, which is composed of the generator group and the carrier group. Each record in an authentication group is represented by t_i.

The length of generator group is denoted as N, which indicates the count of records of the generator group. Besides, the length of the carrier group is denoted as M. How to decide the values of N and M is introduced in the following sections. A record of WSNs data stream includes common information (e.g., data collection time) and several items of collected data (e.g., temperature).

For the tampering detection of an authentication group, the CRC code of the generator group is calculated first as a watermark, and then it is reversibly embedded into the corresponding carrier group using the prediction error-based histogram shifting algorithm [24]. At the decoding end, the integrity of the authentication group is judged according to the comparison result between the watermark extracted from each carrier group and the CRC code computed by the corresponding generator group.

C. WATERMARK GENERATION

Firstly, in a generator group, the j^{th} collected item of the i^{th} record is denoted as $t_{i,j}$, and then each $t_{i,j}$ involving its decimal and integer parts is transformed into binary bits. All the binary bits of the j^{th} collected item of the total records are combined into a binary sequence. If a record has P collected items, then P binary sequences are constructed and are marked as BS. For certain BS_j ($1 \leq j \leq P$), the encoder divides BS_j into several segments, each of which has s bits. Then, the CRC code of each segment is computed, and C_{jm} indicates the CRC code of m^{th} segment *of* BS_j. The detailed computation process of C_{jm} can be referred to in Algorithm 1 below. The count of segments q is calculated by $q = \lceil N/s \rceil$, where the symbol $\lceil x \rceil$ denotes to round x up to the nearest integer, and the final CRC code for BS_j is achieved by the following equation:

$$C_j = C_{j1} \oplus C_{j2} \oplus C_{jm} \oplus \cdots \oplus C_{j(q-1)} \oplus C_{jq}, \qquad (1)$$

where $\oplus$ is XOR operator, $1 \leq j \leq P$, $1 \leq m \leq q$, and C_j is the generated watermark corresponding to the j^{th} collected item of the generator group. The final watermark

generated by the generator group is denoted as C, and $C = [C_1\ C_2\ \cdots C_j \cdots C_{P-1} C_P]$.

Algorithm 1 Watermark Generation

Input: P binary sequences BS
Output: Final watermark C

```
for j = 1 : P
    BS_j is divided into q segments, and q = ⌈N/s⌉;
    for m = 1 : q
    BS_jm is represented as a polynomial. For example,
     '1011001' is expressed as x^6 + x^4 + x^3 + 1;
    BSS_jm = BS_jm ≪ L, where L is a parameter
     controlling the length of the CRC code;
    BSS_jm is regarded as a dividend;
    The polynomial x^L + x^(L-2) + x^(L-3) + 1 is taken as a
     divisor.
    The module-2 division between the dividend and
     divisor is used to obtain remainder C_jm with L-bits;
  end
  C_j = C_j1 ⊕ C_j2 ⊕ C_jm ⊕ ··· ⊕ C_j(q-1) ⊕ C_jq
end
C = [C_1 C_2 ··· C_j ··· C_{P-1} C_P].
```

D. WATERMARK EMBEDDING ·

Assuming that z_i is a record of a carrier group, z_{i+1} *and* z_{i+2} are two neighbor records of z_i, and $z_{i,j}$ indicates the j^{th} collected item of z_i, then the prediction value of $z_{i,j}$ is denoted by $\hat{z}_{i,j}$, which is calculated as follows:

$$\hat{z}_{i,j} = roundn(\frac{z_{i+1,j} + z_{i+2,j}}{2}, -2), \tag{2}$$

where *roundn*(x, -2) denotes rounding off x and keeping two decimals.

Then, the difference $e_{i,j}$ between $z_{i,j}$ and $\hat{z}_{i,j}$, also called prediction error, is computed by Eq. (3).

$$e_{i,j} = z_{i,j} - \hat{z}_{i,j} \tag{3}$$

In Figure 4, an example of a PEH is presented, from which it can be observed that the middle part of the PEH is a smooth region and the two sides of the PEH is a fluctuation region. The change on a smooth region is easier to introduce in visual perception than that on a fluctuation region. In other words, embedding a watermark into a fluctuation region may attract less attention from attackers compared with embedding the watermark into a smooth region [25]. Therefore, the fluctuation region is a better choice for embedding the authentication information than the smooth region.

The prediction errors $e_{i,j}$ at the two sides of the PEH have different values, from which an initial beginning parameter $T_{m,j}$ with a lower absolute value is selected by Eq. (4). Moreover, the end parameter $T_{p,j}$ is decided according to Eq. (5), where $hist(E_j)$ denotes the count of j^{th} collected item with the prediction error value of E_j in a carrier group, and *capacity* is the bit number of the embedded watermark. From Eq. (5),

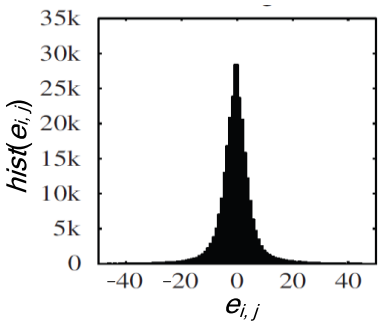

FIGURE 4. An example of a prediction-error histogram.

it can be concluded that the choice of $T_{p,j}$ value can satisfy the capacity demand of the embedded watermark.

$$T_{m,j} = \min(\left|\min(e_{i,j})\right|, \max(e_{i,j})) \tag{4}$$

$$\begin{cases} \text{maximize } T_{p,j} \in (0, 1, 2, \ldots, T_{m,j}) \\ subject\ to\ (\sum_{E_j=-T_{m,j}}^{-T_{p,j}} hist(E_j) \\ \quad + \sum_{E_j=T_{p,j}}^{T_{m,j}} hist(E_j)) > capacity \end{cases} \tag{5}$$

Let b be a bit of watermark, which is embedded into the fluctuation region of PEH using Eq. (6).

$$e'_{i,j} = \begin{cases} e_{i,j} + u, & e_{i,j} > E_j \\ e_{i,j} - u, & e_{i,j} < -E_j \\ e_{i,j} + u, & e_{i,j} = E_j \& b = 1 \\ e_{i,j}, & e_{i,j} = E_j \& b = 0 \\ e_{i,j} - u, & e_{i,j} = -E_j \& b = 1 \\ e_{i,j}, & e_{i,j} = -E_j \& b = 0 \\ e_{i,j}, & else \end{cases} \tag{6}$$

Here, $e'_{i,j}$ is the modified prediction error after embedding the watermark. It can be observed from Eq. (6) that the prediction errors larger than E_j or less than -E_j are shifted by increasing the value of u or decreasing the value of u, and the prediction errors in the smooth region remain unchanged. The value of u can be decided in terms of the type of collected data. In our experiments, the collected data, for example, the humidity, kept two decimals. Therefore, for decreasing the difference between the original and modified prediction errors, the value of u is selected as 0.01, which ensures the embedded WSNs data have only a small change compared with the original WSNs data. Finally, the embedded item value $z'_{i,j}$ is calculated by Eq. (7).

$$z'_{i,j} = e'_{i,j} + \hat{z}_{i,j} \tag{7}$$

It should be noted that the prediction errors of the two corresponding items of the neighbor records $z_{i+1,j}$ *and* $z_{i+2,j}$ of $z_{i,j}$ are not counted and not used to embed the watermark. Besides, $z_{i+1,j}$ *and* $z_{i+2,j}$ are only utilized to calculate the

prediction value of $z_{i,j}$. Moreover, to recover the original authentication group at the decoding end, the parameters $T_{m,j}$ and $T_{p,j}$ as side information need to be embedded by replacing the first forty-two least significant bits (LSBs) of the carrier group. Here, we adopt fourteen and seven bits to save the integer part and decimal part of $T_{m,j}$, respectively. The bit number to save $T_{p,j}$ is the same as that of $T_{m,j}$. Furthermore, for totally restoring the carrier group, the original first forty-two LSBs of the corresponding j^{th} collected item of the carrier group need to be embedded into the fluctuation region of PEH together with the watermark by Eq. (6).

It should be noted that the watermark C is not embedded as a whole at one time. Let L be the bit number of the CRC code C_j, and it is easily known that, for j^{th} collected item in an authentication group, the information to be embedded is composed of the watermark C_j and the first forty-two LSBs of the corresponding j^{th} collected item of the carrier group with a length of $(L+42)$ bits. To make the description clear, we adopt W_j to represent the $(L+42)$ bits of embedded information, and W_j is the authentication information. To assure the total embedding of W_j, the length of the carrier group, namely, the count of records of the carrier group M, is set as $(L+42) \times 9 + 42$. Moreover, for the convenience of detecting the carrier group at the decoding end, a record as a string "000000" is added as the last record of the carrier group, and the notation EOC is used to denote the record. Therefore, each carrier group includes $(L+42) \times 9 + 43$ records.

It should be noted that during the information embedding process according to Eq. (6), the value of E_j is taken decreasingly from $T_{m,j}$ to $T_{p,j}$ with a step size of -0.01; otherwise, the data recovery at the decoding end will not be achieved if the value of E_j is taken increasingly. The detailed embedding procedure of authentication information W may be referred to in Algorithm 2.

E. WATERMARK EXTRACTION AND INTEGRITY AUTHENTICATION

At the decoding end, first, the first forty-two LSBs of the carrier group are extracted to obtain the parameters $T_{m,j}$ and $T_{p,j}$. Then, the modified prediction error $e'_{i,j}$ is calculated by Eq. (8).

$$e'_{i,j} = z'_{i,j} - \hat{z}_{i,j} \tag{8}$$

Then, according to Eq. (9), the embedded information is extracted, where E_j changes from $T_{p,j}$ to $T_{m,j}$.

$$b = \begin{cases} 0, & e'_{i,j} = E_j \quad || \quad e'_{i,j} = -E_j \\ 1, & e'_{i,j} = E_j + u \quad || \quad e'_{i,j} = -E_j - u \end{cases} \tag{9}$$

The original prediction error $e_{i,j}$ is recovered by Eq. (10).

$$e_{i,j} = \begin{cases} e'_{i,j} - u, & e'_{i,j} > E_j + u \\ e'_{i,j} + u, & e'_{i,j} < -E_j - u \\ E_j, & e'_{i,j} = E_j + u \; || e'_{i,j} = E_j \\ -E_j, & e'_{i,j} = -E_j - u \; || e'_{i,j} = -E_j \\ e'_{i,j}, & else \end{cases} \tag{10}$$

Algorithm 2 Embedding

Input: Authentication information W, T_m, T_p, e, $\hat{z}$
Output: Embedded carrier group z'

```
for j = 1: P
  for E_j = T_{m,j}: −0.01: T_{p,j}
    elen = size(e, 1);
    h = 1;
    for i = 1: elen
      if e_{i,j} > E_j
        e'_{i,j} = e_{i,j} + u;
      else if e_{i,j} < - E_j
          e'_{i,j} = e_{i,j}- u;
        else if e_{i,j} == E_j & W_{h,j} == 1
            e'_{i,j} = e_{i,j} + u;
            h = h + 1;
          else if e_{i,j} == E_j & W_{h,j} == 0
              e'_{i,j} = e_{i,j};
              h = h + 1;
            else if e_{i,j} == - E_j & W_{h,j} == 1
                e'_{i,j} = e_{i,j}- u;
                h = h + 1;
              else if e_{i,j} == - E_j & W_{h,j} == 0
                  e'_{i,j} = e_{i,j};
                  h = h + 1;
                else e'_{i,j} = e_{i,j};
              end
            end
          end
        end
      end
      z'_{i,j} = e'_{i,j} + ẑ_{i,j}
    end
  end
end
```

Next, the original WSNs data item is restored by Eq. (11).

$$z_{i,j} = \hat{z}_{i,j} + e_{i,j} \tag{11}$$

Finally, the first forty-two LSBs of the corresponding collected item of the carrier group extracted by Eq. (9) is written back so that all records of a carrier group are completely restored. The concrete information extraction and data restoration procedure may be referred to in Algorithm 3, in which the value of E_j is taken increasingly from $T_{p,j}$ to $T_{m,j}$ with a step size of 0.01 to assure the exact information extraction and data restoration.

After the embedded authentication information is extracted, the decoder can confirm whether the transmitted data for the j^{th} collected item in this authentication group have been tampered in terms of the comparison between the authentication information W generated by the generator group and the extracted authentication information W'. The concrete procedure may be referred to in Algorithm 4, where

the extracted authentication information W' should first be taken inversely because the bit array of W' extracted by Algorithm 3 is the inverse array of the embedded authentication information.

Algorithm 3 Extraction and Restoration

Input: T_m, T_p, e', $\hat{z}$
Output: Restored carrier group z, extracted authentication information W'

```
for j = 1: P
  for E_j = T_{p,j}: 0.01: T_{m,j}
    elen = size(e',1);
    h = 1;
    for i = elen: -1: 1
      if e'_{i,j} > E_j + u
        e_{i,j} = e'_{i,j} - u;
      else if e'_{i,j} < -E_j - u
          e_{i,j} = e'_{i,j} + u;
        else if e'_{i,j} = E_j + u
            e_{i,j} = E_j;
            W'_{h,j} = 1;
            h = h + 1;
          else if e'_{i,j} = E_j
              e_{i,j} = E_j;
              W'_{h,j} = 0;
              h = h + 1;
            else if e'_{i,j} = -E_j - u
                e_{i,j} = -E_j;
                W'_{h,j} = 1;
                h = h + 1;
              else if e'_{i,j} = -E_j
                  e_{i,j} = -E_j;
                  W'_{h,j} = 0;
                  h = h + 1;
                else e_{i,j} = e'_{i,j};
                end
              end
            end
          end
        end
      end
      z_{i,j} = ẑ_{i,j} + e_{i,j}
    end
  end
end
```

F. COMPLEXITY ANALYSIS

The time cost of the proposed algorithm mainly consists of three parts: the CRC code computation, the embedding and extraction of authentication information, and tampering detection. According to Algorithm 1, the time consumption of the CRC code computation is mainly decided by the size of the CRC code L, the count of collected items P, and the count of segments q. Thus, the time complexity for the CRC code computation is $O(P \times q \times L)$. Assuming that the maximum count from $T_{p,j}$ to $T_{m,j}$ is CT, then, in terms of Algorithm 2 and Algorithm 3, it can be concluded that the time complexity of the embedding and extraction of authentication information is $O(P \times CT \times \text{elen})$. Moreover, the time complexity of tampering detection is inferred to be $O(P \times L)$ by Algorithm 4. In summary, the total time complexity is $O(P \times L \times \text{elen})$.

Algorithm 4 Tampering Detection

Input: CRC code C generated by the generator group, extracted authentication information W'
Output: Tampering flag TF.

```
1.  for j = 1: P
2.    for i = 1: L + 42
3.      Winv_{i,j} == W'_{L+43-i,j};
4.    end
5.    C'_j = Winv_{i,j}(1 : L);
6.    if strcmp(C_j, C'_j) == 0
7.      disp ('The j^th collected item in this
              authentication group has been tampered');
8.      TF_j = 1;
9.    else
10.     TF_j = 0;
11.   end
12. end
```

III. EXPERIMENTAL RESULTS AND ANALYSIS

A. EXPERIMENTAL DATA

In our simulation experiments, real WSNs data from Intel Berkeley Lab [26] were adopted. A series of sensor nodes, shown in Figure 5, were deployed in the lab to periodically gather the current temperature, humidity, light, and voltage. The used experimental data stream consisted of 10,000 records, and each record was composed of eight items, including the date, time stamp, epoch, sensor node number, and four other kinds of gathered data, as shown in Figure 6. In this case, an epoch was a monotonically increasing sequence number from each mote.

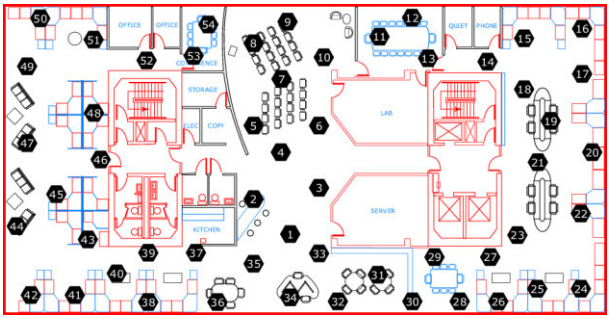

FIGURE 5. Fifty-four sensors deployed in the Intel Berkeley Lab.

Authentication tests were mainly conducted for the gathered data. Considering the practical significance of the

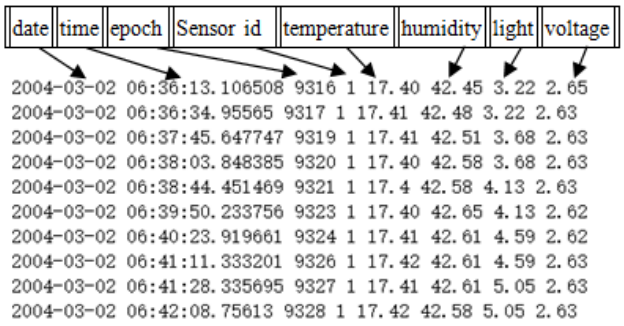

FIGURE 6. Original WSNs data from the Intel Berkeley Lab.

WSNs data, two decimal places of original gathered data were retained by a rounding operation in our experiments. Meanwhile, some disorder records, for example, certain records with an empty gathered item value, were deleted before further data processing. The experimental environment was MATLAB 2012Rb with a CPU I7-4700.

B. SETUP OF KEY PARAMETERS

The size of the generator group in an authentication group is denoted as N, which is taken as a dynamic value for data security. Considering the actual encoding efficiency, the value of N cannot be too small. Meanwhile, the value of N cannot be too large because of the resource limitation of WSNs. After making a comprehensive consideration, the range of the N value was defined as [50, 150]. For a given generator group, the value of N was randomly decided by a key. Furthermore, for avoiding a large false negative rate (FNR) that is analyzed in Section D.4, the size of the CRC code L, namely, the number of bits of CRC code for any one of the items in a generator group cannot take too small a value. Besides, considering the computation cost of the CRC code, it was a fit choice that the value of L was defined as 32.

C. INVISIBILITY AND REVERSIBILITY TEST

Firstly, the invisibility test was conducted. Figure 7 shows an episode of the original WSNs data stream before embedding, and the corresponding embedded data is presented in Figure 8. By comparing the data listed in Figure 7 and Figure 8, only very small differences can be found. For the four kinds of gathered data in a carrier group, the mean square errors (MSEs) between the original and embedded data were 4.0594e-06, 4.2430e-06, 6.5523e-06, and 4.7808e-06, respectively. The computation equation for MSE is defined as follows:

$$MSE = \frac{1}{m}\sum_{i=1}^{m}(d_i - d_i')^2, \tag{12}$$

where d_i and d' indicate the original value and embedded value, respectively; and m is the count of total records in a carrier group. Because the item value only increases or decreases by a value of 0.01 when embedding a bit '1' and the item value does not change when embedding a bit '0,' the mean change of the embedded data is very small. The

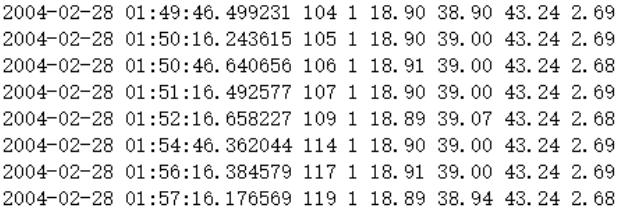

FIGURE 7. Original WSNs data before embedding.

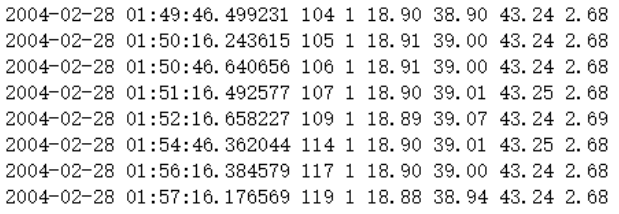

FIGURE 8. Embedded WSNs data.

obtained MSEs with small values indicate that the data stream embedded by the proposed scheme had good invisibility.

Next, a reversibility test of the proposed scheme without being attacked was conducted. Figure 8 shows the restored version of an episode of the original WSNs data stream shown in Figure 6, where any differences could not be found between the original data and the restored data. The indicator, that is, the MSE value between all original WSNs data stream and all the corresponding recovered data streams, was calculated to be 0 by Eq. (12), which demonstrates that the proposed scheme is reversible.

D. TAMPERING DETECTION TEST

Tampering experiments, included insertion, deletion, and modification, were conducted to indicate whether the proposed scheme could detect these types of tampering attacks.

1) INSERTION ATTACK

Firstly, we inserted a record in the generator group of an authentication group in the embedded WSNs data stream, and an example is presented in Figure 10. At the decoding end, the decoding software could detect and report that this authentication group including four items had been tampered, which is shown in Figure 10 (c). After inserting a record in the generator group, at the decoding end, the CRC codes for every item calculated by the modified generator group changed compared with original corresponding CRC codes extracted from the carrier group. Thus, tampering could be correctly detected by the differences between the generated CRC codes and the extracted CRC codes. Furthermore, an example for inserting a record in a carrier group is shown in Figure 11, and a similar detection result as Figure 10 (c) was achieved.

2) MODIFICATION ATTACK

Detections for all kinds of modification attacks were conducted. In Figure 12 (a), a section of a generator group is presented, and Figure 12 (b) shows the section with a modified item in a record listed in Figure 12 (a). The detection result in Figure 12 (b) is listed in Figure 12 (d), where it

2004-02-28 01:49:46.499231 104 1 18.90 38.90 43.24 2.69
2004-02-28 01:50:16.243615 105 1 18.90 39.00 43.24 2.69
2004-02-28 01:50:46.640656 106 1 18.91 39.00 43.24 2.68
2004-02-28 01:51:16.492577 107 1 18.90 39.00 43.24 2.69
2004-02-28 01:52:16.658227 109 1 18.89 39.07 43.24 2.68
2004-02-28 01:54:46.362044 114 1 18.90 39.00 43.24 2.69
2004-02-28 01:56:16.384579 117 1 18.91 39.00 43.24 2.69
2004-02-28 01:57:16.176569 119 1 18.89 38.94 43.24 2.68

FIGURE 9. Restored WSNs data.

2004-02-28 01:09:22.323858 23 1 19.16 38.87 45.07 2.68
2004-02-28 01:09:46.109598 24 1 19.16 38.80 45.07 2.68
2004-02-28 01:10:16.6789 25 1 19.14 38.83 45.07 2.69
2004-02-28 01:10:46.250524 26 1 19.14 38.87 45.07 2.68
2004-02-28 01:11:46.941288 28 1 19.14 38.94 45.07 2.69
2004-02-28 01:12:46.251377 30 1 19.13 38.90 45.07 2.68
2004-02-28 01:14:16.63127 33 1 19.11 38.80 45.07 2.69
2004-02-28 01:14:46.569352 34 1 19.11 38.87 45.07 2.69
2004-02-28 01:15:16.649556 35 1 19.10 39.00 45.07 2.69

(a) A section of a generator group

2004-02-28 01:09:22.323858 23 1 19.16 38.87 45.07 2.68
2004-02-28 01:09:46.109598 24 1 19.16 38.80 45.07 2.68
2004-02-28 01:10:16.6789 25 1 19.14 38.83 45.07 2.69
2004-02-28 01:10:46.250524 26 1 19.14 38.87 45.07 2.68
2004-02-28 01:10:46.667512 27 1 19.14 38.89 45.07 2.69
2004-02-28 01:11:46.941288 28 1 19.14 38.94 45.07 2.69
2004-02-28 01:12:46.251377 30 1 19.13 38.90 45.07 2.68
2004-02-28 01:14:16.63127 33 1 19.11 38.80 45.07 2.69
2004-02-28 01:14:46.569352 34 1 19.11 38.87 45.07 2.69
2004-02-28 01:15:16.649556 35 1 19.10 39.00 45.07 2.69

(b) Inserting a record in (a)

The all items in the 1st authentication group have been tampered.

(c) Detection result on Figure 10 (b)

FIGURE 10. Insertion attack in a generator group of the embedded WSNs data stream.

2004-02-29 07:04:49.339355 3614 1 17.47 43.38 57.03 2.65
2004-02-29 07:05:20.270597 3615 1 17.47 43.41 57.03 2.66
2004-02-29 07:05:51.907391 3616 1 17.47 43.45 57.03 2.66
2004-02-29 07:06:56.029521 3618 1 17.47 43.35 57.03 2.66
2004-02-29 07:08:21.523644 3621 1 17.46 43.38 60.71 2.67
2004-02-29 07:09:24.934961 3623 1 17.46 43.38 60.71 2.66
2004-02-29 07:10:19.445668 3625 1 17.46 43.31 60.71 2.66
2004-02-29 07:10:55.040249 3626 1 17.47 43.35 64.40 2.66
2004-02-29 07:11:52.062192 3628 1 17.46 43.31 64.4 2.66

(a) A section of a carrier group

2004-02-29 07:04:49.339355 3614 1 17.47 43.38 57.03 2.65
2004-02-29 07:05:20.270597 3615 1 17.47 43.41 57.03 2.66
2004-02-29 07:05:51.907391 3616 1 17.47 43.45 57.03 2.66
2004-02-29 07:06:56.029521 3618 1 17.47 43.35 57.03 2.66
2004-02-29 07:07:26.128745 3620 1 17.47 43.37 58.15 2.67
2004-02-29 07:08:21.523644 3621 1 17.46 43.38 60.71 2.67
2004-02-29 07:09:24.934961 3623 1 17.46 43.38 60.71 2.66
2004-02-29 07:10:19.445668 3625 1 17.46 43.31 60.71 2.66
2004-02-29 07:10:55.040249 3626 1 17.47 43.35 64.40 2.66
2004-02-29 07:11:52.062192 3628 1 17.46 43.31 64.4 2.66

(b) Inserting a record in (a)

FIGURE 11. Insertion attack in a carrier group of the embedded WSNs data stream.

can be observed that the tampering for humidity item was detected and the other items passed the integrity authentication. Then, by the modification for two items shown in Figure 12(c) and the corresponding detection result shown in Figure 12 (e), it can be concluded that the tampering detection by our method could achieve concrete precision

2004-02-28 01:09:22.323858 23 1 19.16 38.87 45.07 2.68
2004-02-28 01:09:46.109598 24 1 19.16 38.80 45.07 2.68
2004-02-28 01:10:16.6789 25 1 19.14 38.83 45.07 2.69
2004-02-28 01:10:46.250524 26 1 19.14 38.87 45.07 2.68
2004-02-28 01:11:46.941288 28 1 19.14 38.94 45.07 2.69
2004-02-28 01:12:46.251377 30 1 19.13 38.90 45.07 2.68
2004-02-28 01:14:16.63127 33 1 19.11 38.80 45.07 2.69
2004-02-28 01:14:46.569352 34 1 19.11 38.87 45.07 2.69
2004-02-28 01:15:16.649556 35 1 19.10 39.00 45.07 2.69

(a) A section of a generator group

2004-02-28 01:09:22.323858 23 1 19.16 38.87 45.07 2.68
2004-02-28 01:09:46.109598 24 1 19.16 38.80 45.07 2.68
2004-02-28 01:10:16.6789 25 1 19.14 38.84 45.07 2.69
2004-02-28 01:10:46.250524 26 1 19.14 38.87 45.07 2.68
2004-02-28 01:11:46.941288 28 1 19.14 38.94 45.07 2.69
2004-02-28 01:12:46.251377 30 1 19.13 38.90 45.07 2.68
2004-02-28 01:14:16.63127 33 1 19.11 38.80 45.07 2.69
2004-02-28 01:14:46.569352 34 1 19.11 38.87 45.07 2.69
2004-02-28 01:15:16.649556 35 1 19.10 39.00 45.07 2.69

(b) Modifying an item of a record in (a)

2004-02-28 01:09:22.323858 23 1 19.16 38.87 45.07 2.68
2004-02-28 01:09:46.109598 24 1 19.16 38.80 45.07 2.68
2004-02-28 01:10:16.6789 25 1 19.14 38.83 45.07 2.69
2004-02-28 01:10:46.250524 26 1 19.14 38.87 45.06 2.68
2004-02-28 01:11:46.941288 28 1 19.14 38.94 45.07 2.69
2004-02-28 01:12:46.251377 30 1 19.13 38.90 45.07 2.68
2004-02-28 01:14:16.63127 33 1 19.11 39.80 45.07 2.69
2004-02-28 01:14:46.569352 34 1 19.11 38.87 45.07 2.69
2004-02-28 01:15:16.649556 35 1 19.10 39.00 45.07 2.69

(c) Modifying two items of two records in (a)

The item of humidity in the 1st authentication group has been tampered, and the other items have passed the integrity authentication.

(d) Detection result on Figure 12 (b)

The items of humidity and light in the 1st authentication group have been tampered, and the other items have passed the integrity authentication.

(e) Detection result on Figure 12 (c)

FIGURE 12. Modification attack in a generator group of the embedded WSNs data stream.

of one or more items for an authentication group. At the encoding end, the CRC code of each item for a generator group was computed and embedded into the carrier group. At the decoding end, the CRC code computed by the modified item in the generator group had to be inconsistent with the corresponding original CRC code extracted from the carrier group. Therefore, we could present a correct judgement for the integrity certification of a certain item in an authentication group.

Moreover, experiments were conducted for modifying one or more items of a record in the carrier group and simultaneously modifying items in both the generator group and carrier group, respectively. Similar detection results to Figure 12 (d, e) could be achieved, which are not listed here due to space constraints.

3) DELETION ATTACK

Detections for all kinds of deletion attacks were conducted. In Figure 13 (a), a section of a generator group is presented, and Figure 13 (b) shows the section with a deleted record in Figure 13 (a). The detection result on Figure 13 (b) is listed in Figure 13 (c), where it can be observed that tampering had been detected.

Furthermore, experiments were conducted for deleting one or more records in the carrier group and simultaneously

```
2004-02-28 01:09:22.323858 23 1 19.16 38.87 45.07 2.68
2004-02-28 01:09:46.109598 24 1 19.16 38.80 45.07 2.68
2004-02-28 01:10:16.6789 25 1 19.14 38.83 45.07 2.69
2004-02-28 01:10:46.250524 26 1 19.14 38.87 45.07 2.68
2004-02-28 01:11:46.941288 28 1 19.14 38.94 45.07 2.69
2004-02-28 01:12:46.251377 30 1 19.13 38.90 45.07 2.68
2004-02-28 01:14:16.63127 33 1 19.11 38.80 45.07 2.69
2004-02-28 01:14:46.569352 34 1 19.11 38.87 45.07 2.69
2004-02-28 01:15:16.649556 35 1 19.10 39.00 45.07 2.69
```

(a) A section of a generator group

```
2004-02-28 01:09:22.323858 23 1 19.16 38.87 45.07 2.68
2004-02-28 01:09:46.109598 24 1 19.16 38.80 45.07 2.68
2004-02-28 01:10:16.6789 25 1 19.14 38.83 45.07 2.69
2004-02-28 01:10:46.250524 26 1 19.14 38.87 45.07 2.68
2004-02-28 01:11:46.941288 28 1 19.14 38.94 45.07 2.69
2004-02-28 01:14:16.63127 33 1 19.11 38.80 45.07 2.69
2004-02-28 01:14:46.569352 34 1 19.11 38.87 45.07 2.69
2004-02-28 01:15:16.649556 35 1 19.10 39.00 45.07 2.69
```

(b) Deleting a record in (a)

```
The all items in the 1st authentication group have been tampered.
```

(c) Detection result on Figure 13 (b)

FIGURE 13. Deletion attack in a generator group of the embedded WSNs data stream.

deleting one or more records in the generator group and carrier group, respectively. Similar detection results to Figure 13 (c) could be achieved, which are not listed here due to space constraints.

4) ANALYSIS OF THE FALSE-POSITIVE RATE AND FALSE NEGATIVE RATE

Firstly, the FPR is theoretically analyzed. If tampering attacks, including insertion, modification, and deletion operations, are not conducted, then every authentication group can be divided correctly at the decoding end. Because the generator group and the carrier group from an authentication group are not tampered, the CRC code calculated by the generator group must be the same as that extracted by the carrier group. That is to say, the decoding end presents the result of passing integrity certification for an authentication group without being tampered, which indicates that the FPR is zero.

Next, the FNR for the three categories of tampering is analyzed. For an authentication group, the first category of tampering is that only its generator group is tampered and its carrier group is unchanged, which is further divided into two subcategories including a modified generator group with an unchanged carrier group and an inserted or deleted generator group with an unchanged carrier group. For the first subcategory, in terms of the principle of CRC generation, the CRC code generated by the modified generator group must be inconsistent with the original CRC code extracted by the unchanged carrier group. Therefore, in this case, the FNR is zero. For the second subcategory of inserting one or more records in the generator group, the last one or more records of the original generator group will be considered wrongly as a part of the carrier group by the decoder. In our experiments, the number of bits of CRC code was defined as 32, and thus it was easy to know that the probability with two consistent CRC codes, where one is generated by the inserted generator group and the other is extracted by the misidentified carrier group, is $1/2^{32}$, which indicates the FNR is $1/2^{32}$ in this case. Similarly, for the other case of the deleted generator group with an unchanged carrier group, it can be concluded that the FNR is also $1/2^{32}$.

For an authentication group, the second category of tampering is that only a certain type of item value in its carrier group is tampered and the corresponding type of the item value in its generator group is unchanged. The CRC code generated by the generator group is same as the original CRC code. The CRC code extracted by the tampered carrier group is consistent with the original CRC code with a probability of $1/2^{32}$, that is, the FNR is $1/2^{32}$ in the case of the second category of tampering.

The third category of tampering is that the certain type of item values in the generator group and the corresponding type of the item value in the carrier group are simultaneously tampered. In this case, the two CRC codes, where one is generated by the generator group and the other is extracted by the carrier group, all may not be same as the original CRC code. The probability that the two CRC codes are fully consistent is $1/2^{32}$. Therefore, the FNR is $1/2^{32}$ for the third category of tampering. Table 1 lists the FPR and FNR of different categories of tampering for an authentication group.

TABLE 1. FPR and FNR Analysis for an Authentication Group.

Tampering category	FPR	FNR
Not being tampered	zero	–
First subcategory of the first category of tampering	–	zero
Second subcategory of the first category of tampering	–	$1/2^{32}$
Second category of tampering	–	$1/2^{32}$
Third category of tampering	–	$1/2^{32}$

Next, practical integrity certification tests for the above-mentioned three categories of tampering attacks were performed. Each category of tampering attack involved modification, insertion, and deletion. If the tampering operation was not be detected by the decoder, then it is thought that the integrity certification failed; otherwise, the integrity certification was successful. Each category of tampering attack used 50 random samples, and Table 2 shows the detection failure rate and detection success rate for three categories of tampering attacks with the adopted WSNs data stream.

TABLE 2. Detection Failure Rate and Detection Success Rate for Three Categories of Tampering Attacks.

Tampering category	Detection failure rate	Detection succeed rate
First category of tampering	0	100%
Second category of tampering	0	100%
Third category of tampering	0	100%

TABLE 3. Comparisons Among Several up-to-date RDH-Based WSNs Data Authentication Schemes.

Evaluation indicators	Shi's scheme [19]	Wang's scheme [20]	Wu's scheme [21]	Jiang's scheme [22]	Proposed scheme
Embedding method	Prediction error expansion	Prediction error expansion	Odd-even invariability	Difference expansion	Prediction error histogram shifting
MSE between original and embedded item values	>= 0.5	>= 0.5	>= 0.5	>= 0.5	<= e-5
Embedding region	Smooth and fluctuation regions	Smooth and fluctuation regions	Smooth and fluctuation regions	Smooth and fluctuation regions	Only fluctuation region
Watermark generation strategy	Hash	Hash	CRC	Chaos sequence	CRC
Division of authentication group	Dynamic	Dynamic	Static	Static	Dynamic
FPR	Low	Low	High	High	Low
Category of authenticated item	Only one category of item	Only one category of item	Only one category of item	Only one category of item	Multiple categories of items
Data type of authenticated item	Integer	Integer	Integer	Integer	Integer and decimal

E. COMPARISION WITH UP TO DATE ALGORITHMS

Comparisons among several up to date RDH-based WSNs data authentication schemes, including Shi's scheme [19], Wang's scheme [20], Wu's scheme [21], Jiang's scheme [22], and the proposed scheme, are provided. In Table 3, the differences between multiple evaluation indicators for these schemes are listed. Table 3 shows that the MSE between the original and embedded item values in the proposed method was far less than those in the other four schemes; in other words, the rate between the MSE achieved by the proposed scheme and that achieved by the other schemes was less than 0.002%, which indicates that the invisibility of the proposed method was better than the other schemes. Moreover, in the proposed scheme, the watermark embedding region was only located in the fluctuation region of the PEH. In contrast, the other four schemes embedded the watermark into both smooth and fluctuation regions, which may lead to more visual perception and more attention from attackers than the proposed scheme. In Shi's scheme [19] and Wang's scheme [20], watermark information is generated by calculating the hash value. However, the computation efficiency of the hash value was far less than that of chaos sequence adopted in Jiang's scheme [22] and that of the CRC value adopted in Wu's scheme [21] and the proposed scheme. Experimental results demonstrate that the average rate between the computation time of CRC code and that of the MD5 hash value for a data stream with the length of 56 chars was 0.054%. That is to say, a higher time complex was achieved in Shi's scheme [19] and Wang's scheme [20].

For the division of the authentication group, Wu's scheme [21] and Jiang's scheme [22] use a static strategy, namely, the group size is fixed. Therefore, when a record is inserted or deleted by attacker in an authentication group, all the following authentication groups will be identified improperly, and thus that the induced FPR is higher than the proposed scheme and the other two schemes. In extreme situations, if a record is inserted into the first group by an attacker and all the following groups are not attacked, then the FPR of the schemes using a static group strategy will almost be 100%. In contrast, for the same situation, the FPR of the schemes using a dynamic group strategy will almost be 0%. The proposed scheme aims at the authentication for multiple categories of items; however, the other four schemes only consider the authentication for one category of item. Therefore, in comparison, the proposed scheme has better practicability and universality. Finally, the data type of authenticated item is only an integer in the other four schemes; in contrast, the proposed scheme is suitable for more data types involving integers and decimals. In summary, it can be concluded that the proposed scheme is superior to the other four up-to-date RDH-based authentication schemes.

IV. CONCLUSION

Based on the CRC code and PEH, a reversible multiple items authentication scheme for WSNs is proposed in this article. To enhance the universality of WSNs authentication, the CRC code of the generator group as an authentication watermark is composed of the CRC codes of all the collected items in the generator group by which the authentication of multiple items in WSNs can be implemented. Furthermore, in the embedding phase, to increase the imperceptibility of embedded items in the carrier group, only the fluctuation region of PEH is selected to embed the authentication watermark. The proposed scheme can detect these tampering attacks, including insertion, deletion, and modification attacks. In future work, further degrading the time complexity will be considered.

REFERENCES

[1] C. Zhao, C. Wu, X. Wang, B. Ling, K. Teo, J. Lee, and K. Jung, "Maximizing lifetime of a wireless sensor network via joint optimizing sink placement and sensor-to-sink routing" *Appl. Math. Model.*, vol. 49, pp. 319–337, Sep. 2017.

[2] M. Zareei, C. Vargas-Rosales, R. Villalpando-Hernandez, L. Azpilicueta, M. H. Anisi, and M. H. Rehmani, "The effects of an adaptive and distributed transmission power control on the performance of energy harvesting sensor networks," *Comput. Netw.*, vol. 137, pp. 69–82, Jun. 2018.

[3] C. Karlof and N. Sastry, "TinySec: A link layer security architecture for wireless sensor networks," in *Proc. 2nd Int. Conf. Embedded Netw. Sensor Syst.*, Baltimore, MD, USA, 2004, pp. 162–175.
[4] I. Ullah, N. U. Amin, M. Zareei, A. Zeb, H. Khattak, A. Khan, and S. Goudarzi, "A lightweight and provable secured certificateless signcryption approach for crowdsourced IIoT applications," *Symmetry*, vol. 11, no. 11, p. 1386, 2019.
[5] S.-H. Seo, J. Won, S. Sultana, and E. Bertino, "Effective key management in dynamic wireless sensor networks," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 2, pp. 371–383, Feb. 2015.
[6] C.-M. Chen, Y.-H. Lin, Y.-C. Lin, and H.-M. Sun, "RCDA: Recoverable concealed data aggregation for data integrity in wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 4, pp. 727–734, Apr. 2012.
[7] S. Paramasivam and P. Kalpana, "Novel reversible design of advanced encryption standard cryptographic algorithm for wireless sensor networks," *Wireless Pers. Commun.*, vol. 100, no. 2, pp. 1427–1458, 2018.
[8] A. Perrig, V. Szewczyk, D. Wen, and D. Culler, "SPINS: Security protocols for sensor networks," in *Proc. 7th Annu. Int. Conf. Mobile Comput. Netw.*, Rome, Italy, 2001, pp. 131–141.
[9] S. Zhu and S. Setia, "LEAP+: Efficient security mechanisms for large-scale distributed sensor networks," in *Proc. 10th ACM Conf. Comput. Commun. Secur.*, Washington, DC, USA, 2018, pp. 62–72.
[10] J. Wong, J. Feng, and D. Kirovski, "Security in sensor networks: Watermarking techniques," in *Wireless Sensor Networks*. Boston, MA, USA: Springer, 2004, pp. 305–323.
[11] T.-S. Chen, K.-N. Hou, W.-K. Beh, and A.-Y. Wu, "Low-complexity Compressed-Sensing-Based watermark cryptosystem and circuits implementation for wireless sensor networks," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 11, pp. 2485–2497, Nov. 2019.
[12] H. Guo, Y. Li, and S. Jajodia, "Chaining watermarks for detecting malicious modifications to streaming data," *Inf. Sci.*, vol. 177, no. 1, pp. 281–298, 2007.
[13] I. Kamel and H. Guma, "Simplified watermarking scheme for sensor networks," *Int. J. Internet Protocol Technol.*, vol. 5, no. 1, pp. 101–111, 2019.
[14] I. Kamel and H. Juma, "A lightweight data integrity scheme for sensor networks," *Sensors*, vol. 11, no. 4, pp. 4118–4136, Apr. 2011.
[15] S. Kim, R. Lussi, X. Qu, F. Huang, and H. Kim, "Reversible data hiding with automatic brightness preserving contrast enhancement," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 8, pp. 2264–2271, Aug. 2019.
[16] S. Yi and Y. Zhou, "Separable and reversible data hiding in encrypted images using parametric binary tree labeling," *IEEE Trans. Multimedia*, vol. 21, no. 1, pp. 51–64, Jan. 2019.
[17] X. Li, W. Zhang, X. Gui, and B. Yang, "Efficient reversible data hiding based on multiple histograms modification," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 9, pp. 2016–2027, Sep. 2015.
[18] X. Chen, H. Zhong, and A. Qiu, "Reversible data hiding scheme in multiple encrypted images based on code division multiplexing," *Multimedia Tools Appl.*, vol. 78, no. 6, pp. 7499–7516, Mar. 2019.
[19] X. Shi and D. Xiao, "A reversible watermarking authentication scheme for wireless sensor networks," *Inf. Sci.*, vol. 240, pp. 173–183, Aug. 2013.
[20] C. Wang, A. Wu, and S. Huang, "An energy conserving reversible and irreversible digital watermarking hybrid scheme for cluster-based wireless sensor networks," *J. Internet Technol.*, vol. 19, no. 1, pp. 105–114, 2018.
[21] H. Wu, Y. Chen, and Z. Ji, "Wireless sensor networks authentication algorithm based on CRC and reversible digital watermarking," *Comput. Appl. Softw.*, vol. 33, no. 6, pp. 294–298, 2016.
[22] W. Jiang, Z. Zhang, and J. Wu, "Reversible digital watermarking-based protocol for data integrity in wireless sensor network," *J. Commun.*, vol. 39, no. 3, pp. 1–10, 2018.
[23] X. Liu, Y. Ge, Y. Zhu, and D. Wu, "A lightweight integrity authentication scheme based on reversible watermark for wireless body area networks," *KSII Trans. Internet Inf. Syst.*, vol. 8, no. 12, pp. 4643–4660, 2014.
[24] Y. Yang, W. Zhang, D. Hou, and H. Wang, "Research and prospect of reversible data hiding method with contrast enhancement," *Chin. J. Netw. Inf. Secur.*, vol. 2, no. 4, pp. 12–19, 2016.
[25] Y. Yang, W. Zhang, and N. Yu, "Improving visual quality of reversible data hiding in medical image with texture area contrast enhancement," in *Proc. Int. Conf. Intell. Inf. Hiding Multimedia Signal Process. (IIH-MSP)*, Adelaide, SA, Australia, Sep. 2015, pp. 81–84.
[26] *Intel Lab Data*. Accessed: Jul. 20, 2019. [Online]. Available: http://db.lcs.mit.edu/labdata/labdata.html

GUANGYONG GAO received the Ph.D. degree from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2012.

He is currently a Professor with the School of Computer and Software, Nanjing University of Information Science & Technology. His research interests include computer networks security, multimedia information security, and digital image processing.

BIN WU received the B.S. degree from Shandong Normal University, in 2003, and the M.S. degree from the Dalian University of Technology, in 2009.

He is currently a Lecturer with the School of Information Science and Technology, Jiujiang University. His research interests include database security, network security, and cloud security.

YUXIANG WANG received the Ph.D. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2011.

He is currently a Lecturer with the School of Computer and Software, Nanjing University of Information Science & Technology. His research interests include computer networks security and multimedia information security.

ZHIQIANG ZHAO received the Ph.D. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2018.

He is currently a Lecturer with the School of Information Science and Technology, Jiujiang University. His research interests include visual tracking, computer vision, and image processing.

LIYA XU received the Ph.D. degree from Wuhan University, Wuhan, China, in 2014.

He is currently a Lecturer with the School of Information Science and Technology, Jiujiang University. His research interests include VANET, the Internet of Things, and DTN.

CHAO YIN received the B.E., M.S., and Ph.D. degrees from the Huazhong University of Science and Technology, in 2001, 2005, and 2014, respectively.

He is currently an Associate Professor with the School of Information Science and Technology, Jiujiang University. His research interests include big data, cloud storage, and erasure codes.