IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 30, 2019, accepted September 17, 2019, date of publication October 2, 2019,
date of current version October 17, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2945177

Digital Watermark-Based Independent Individual Certification Scheme in WSNs

YAN XIAO[1] AND GUANGYONG GAO[1,2]
[1]School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China
[2]School of Information and Science Technology, Jiujiang University, Jiujiang 332005, China

Corresponding author: Guangyong Gao (gaoguangyong@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61662039 and Grant 61362032, 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 and Technology (NUIST) under Grant 2019r070, and in part by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) fund.

ABSTRACT With the development of wireless sensor networks (WSNs), data security has gained considerable attention. Since the sensor node is in a special environment with small volume and limited storage space, it is highly likely to be attacked illegally when transmitting the data. This paper proposes a watermarking scheme based on individual certification for sensor nodes. The data is grouped in a sensor node. A watermarking bit is generated by associating several other data items within a group, and is embedded in the group as a space. At the sink node, the extracted watermarking bit is compared with the watermarking bit generated by the same way in the sensor node to determine whether the data item has been tampered. The watermarking bit embedded in the proposed scheme is extremely difficult to be found in the transmission process. Meanwhile, it does not change the original data. The experimental results demonstrate the proposed scheme has superior performance compared with other certification schemes in WSNs.

INDEX TERMS Watermark, individual certification, embedded spaces, data group.

I. INTRODUCTION

Wireless sensor network (WSN) is usually composed of sensor, sink and management nodes. The sensor nodes generally collect specific data in certain environments. Then the data collected from the sensor node is transmitted wirelessly to the sink node. WSNs can be applied in many fields because of the characteristics of sensor nodes and wireless connection. For example, in the military field, WSN can be used to monitor equipment in enemy areas, battlefield conditions in real time and locate targets. In industrial safety, sensor network technology can be used in dangerous work environments. For example, sensor nodes can be placed to monitor the safety status of the work environment and provide security for the staff in coal mines and nuclear power plants.

However, the data can be easily obtained and modified by an attacker in the process. False data would result in incorrect decisions. Therefore, the data should be verified at the sink node. Most approaches of data validation are holistic validation of a group of data. If the watermark does not match, the set of data is discarded [1]–[4].

For example, Kamel and Juma [1] introduced the FWC-D algorithm, which divides the data into groups of fixed size and embeds the watermark into the LSB. The two embedded methods that can be adopted are embedding the watermarking bit into the front group or the next group for transmission. Once the data is tampered with, the entire set of data is rejected by the receiver sensor. Zhang *et al.* [2] proposed a scheme that authenticated the identity of the data and the node by digital watermarking technology. The first step is that the witness node sends the witness watermark w_i and its own serial number of key p to the cluster head. The second step is that the base station verifies the authenticity of the received data. Sun *et al.* [3] found that there would be redundant space for data collection and storage. Therefore, the watermarking bit was embedded into the redundant space and transmitted to the base station with the data. Guan and Chen [4] proposed a solution to embed the whole watermark into the data set. To validate the grouped data, only the change in the watermark is detected, and the grouped data is discarded. In order to avoid destroying the original data, Kamel and Alkoky [5]

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

proposed a data sorting method based on watermark. Thus, the watermarking bit is not embedded into the data. Zhou and Zhang [6] embedded the data and the watermark information together into the hash sequence, which avoided destroying the data transmitted by sensor nodes. The goal is to minimize corruption of data values. Shi and Xiao [7] proposed that in the case of unsuccessful data group validation, the data group could be divided again for double verification. Later, Shi [8] abandoned the concept of grouping and introduced the idea of queue. As each datum passed through the queue, a watermarking bit was generated in the queue and embedded into the data. In this way, the watermark verification problem caused by grouping was avoided. Li *et al.* [9] proposed a new idea and considered the sensor nodes distributed in each environment as the distributed pixels in the image to generate distributed watermarks. This was a reversible solution in which the original data was calculated according to the difference vector. Lalem *et al.* [10] conducted simulation experiments on watermarking bit embedding using the linear interpolation. Hameed *et al.* [11] proposed to utilize the characteristics of the data itself to generate and encrypt the watermark, and then transmit the watermarking bit with the data to the authentication node. Baoyi *et al.* [12] proposed a WSN secure communication solution with less time overhead for electric transmission line based on digital watermark. Wang *et al.* [13] proposed an effective dual-chaining watermark scheme, called DCW. In order to improve the verification of individual data, this paper proposes a new scheme. Compared with some current authentication schemes in WSNs, our main contribution is not only making the watermark embedding process more confidential, but also not changing the original data on the basis of individual authentication.

The remainder of this paper is organized as follows. Section 2 describes the proposed scheme of watermark embedding and extraction process. Section 3 proposes an experimental scheme for analysis. Section 4 analyzes the experimental results. Section 5 presents the conclusions.

II. INDIVIDUAL DATA VERIFICATION SCHEME

For simplicity, assume that the data being transmitted is numeric. A continuous data stream is formed from sensor node to sink node. The data flow from the sensor to the sink node is set as S, and the data collected by the sensor node in the environment is set as a data item s_i. Sensitive data and time collected are included in the data item.

The scheme presented in this paper adopts part of the idea of Shi [14]. The basic process of the proposed scheme is as follows. The sensor node caches a data group with a packet length of N. In this group, M data items are selected for watermark calculation. Each data item participated in the watermark calculation for M times and the resulting watermark value is folded into one bit. Thus, N watermarking bits are generated and embedded into each data group. The data is grouped at the sink node. The watermark is calculated and folded in similar manner as the watermarking bit embedded. Data tampering can be identified by changes in the watermark.

A. WATERMARK GENERATION AND EMBEDDING

In the sensor node, the collected data is processed. A data group is represented by D, including N data items and can be expressed as $D = \{s_i, s_{i+1} \ldots s_{i+N-1}\}$.

Step 1: Generate an $N \times N$ matrix B according to the group length N. Ensure that each row and column has M elements of 1 and the rest of the elements are 0. The B matrix is used to ensure that each data item can participate in M hash operations. Assuming N and M are equal to 4 and 2, respectively, matrix B is as follows:

$$B = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

Step 2: Use D to calculate the watermark. The N items in D are multiplied by B matrix, and $1 \times N$ matrix is obtained. The purpose is to ensure that each element value in $1 \times N$ matrix is associated with M data items.

Step 3: Compute the hash value of each member of a $1 \times N$ matrix, denoted as H, of the candidates using the secure hash function MD5. The binary representation of H is $b_j b_{j-1} \ldots b_0$. Bitwise XOR operation is performed to fold H into a one-bit watermark denoted as w_j. The watermark value of the j^{th} element is represented as w_j. The calculation formula is as follows:

$$w_j = b_j \oplus b_{j-1} \ldots \oplus b_0$$

where $j = 1, 2 \ldots N$. Using the above formula, N bit watermarks can be obtained.

Step 4: Convert the group of data $D = \{s_i, s_{i+1} \ldots .. s_{i+N-1}\}$ to character data $D\_C = \{C(s_i), C(s_{i+1}) \ldots . C(s_{i+N-1})\}$. Where $C(s_i)$ represents the function that converts the double numeric data s_i to character data. If the watermark value is '1', embed a space in the set of data s_i. If the watermark value is '0', do not embed the watermarking bit into the data. Attain the data set $D\_Q = \{Q(s_i), Q(s_{i+1}) \ldots Q(s_{i+N-1})\}$. Where $Q(s_i)$ represents the function that embed watermarks into data items.

In wireless sensor networks, the data collected by sensor nodes in a certain time cycle are not immediately transmitted to the sink nodes because the data must be converted to a character type before the watermark is embedded. Finally, at the sink node, the data is verified. The flow of the embedding model is shown in figure 1.

In the proposed watermark embedding algorithm, zerosones(N, M) represents the $N \times N$ matrix B, in which the number of elements of 1 is M for each row and column. The function, Hash(*data*), generates a fixed-length Hash value for the data. Folding(H) means that the data is folded into a one-bit watermark by bitwise XOR operation. The MATLAB function strcat(a, b) concatenates strings horizontally, while

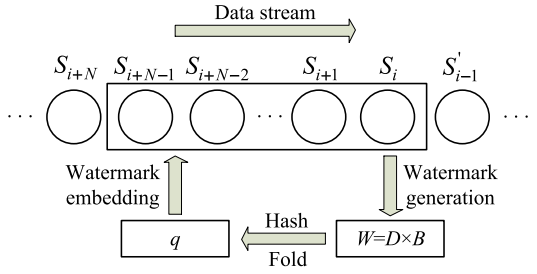


FIGURE 1. Watermark generation and embedding model.

TABLE 1. Embedding watermark algorithm.

Algorithm 1 embedding watermark algorithm
Input: Data group D, the length of a group N, the number of elements of 1 M, Embedded watermark data set D_Q
Output: Watermarked data
B= zerosones(N,M);
//Each data item is calculated M times
W=D*B;
//Calculate the hash value
H=Hash(W);
q= Folding(H);
//Convert and embed watermarks
If q==1
D_Q=strcat(sprintf('%s', D), ' ');
Else
D_Q=sprintf('%s', D));
End

sprintf(*data*) formats the data into string. The algorithm process is described in Table 1.

B. WATERMARK EXTRACTION AND COMPARISON

After the data is received from the sensor nodes, it is grouped in the same way as the watermarking embedding process.

Step 1: Check for a space at the end of the character data. If there is, the watermark '1' is extracted. Otherwise, extract the watermark '0'. Denote the result of judgment as W_q

Step 2: Convert character data to double type. The data set is denoted as D'.

Step 3: Compute the double data to obtain the watermark as the same way as the watermarking embedding process, which is denoted as W_h.

Step 4: Compare W_q with W_h, if they are not equal, they are marked as '1' in the *Res* array, otherwise they are marked as '0'. Thus, N results of the comparisons are obtained in the *Res* array.

The sink node receives the data from the sensor node. If the element in the *Res* array is '1', the corresponding data has been changed. The watermark extraction model is shown in figure 2.

In the watermark extraction algorithm, Str(*data*) is a function that converts the character data to double data. Generation (*data*) means watermark is generated by the same way as the watermarking embedding process. The watermark extraction algorithm is shown in Table 2.

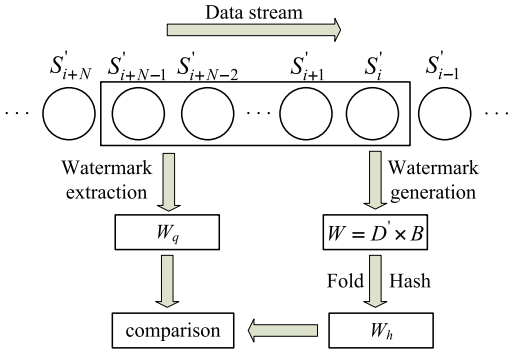


FIGURE 2. Watermark extraction and comparison model.

TABLE 2. Watermark extraction algorithm.

Algorithm 2 watermark extraction algorithm
Input: Data stream D', The length of a group N, Embedded watermark data set D_Q
Output: Watermark comparison results
//Gets the watermark value
For i=1:N
If D_Q(1,i)==' '
W_q=1;
Else
W_q=0;
End
End
//Transform data type
D'=Str(D_Q);
//compute watermark in the same way as the watermarking embedding process
W_h=Generation(D');
// compare watermark
If W_q~=W_h
Res=1;
Else
Res=0;
End

III. EXPERIMENTAL SCHEME ANALYSIS

Two indicators are used to measure the performance of the proposed scheme. While comparing the predicted results with the real results, some errors in judgment are expected. The case where the real result is incorrect but the prediction is correct, is called false negative, while a false positive is when the real result is correct but the prediction is incorrect.

A. THE THRESHOLD T

A grouping Q in the data stream cached by the sink node contains N data items s_i, $s_{i+1} \ldots s_{i+N-1}$. If data item s_i has been tampered with, M watermark bits are associated with s_i in the N watermark bits are generated by calculation in data set Q. Due to the folding operation of hash values, the result of the data item changes with a probability of 1/2. The watermark value may change from '0' to '1' or from '1' to '0', and the change in M bits are distributed independently and identically.

Assume Y is the number of flips in M bits, the probability distribution of Y is as follows:

$$P(Y = k) = \binom{M}{k} \cdot \left(\frac{1}{2}\right)^{M} \tag{1}$$

where $k = 0, 1, 2, 3 \ldots M$.

When k is close to 0, the probability P($Y = k$) is small enough to be ignored. Define a threshold T as the maximum number of flips. If $Y <= T$, the data item s_i is considered unchanged.

B. FALSE NEGATIVE RATE

The false negative rate (FNR) of experiment is the ratio of the tampered data to the tampered data that can be detected. If the number of watermark flips associated with the tampered data is less than T, then the data item is not detected. Assume the number of the misjudged data items and the total number of tampered data items are denoted as X and $Xnumber$, respectively. The false negative rate can be expressed as follows:

$$FNR = \frac{X}{Xnumber} \tag{2}$$

C. FALSE POSITIVE RATE

A data item that has not been tampered with is misjudged as a tampered data item, because the watermarking bits of the data item are affected by the tampered data. Therefore, M watermark bits associated with the data item will have a 1/2 probability of flipping. If the number of flips in the M watermarking bits that are associated the data item is more than T, the data item will be judged as tampered data. Assume the numbers of the data items that has not been tampered with and the misjudged data items are denoted as $Vnumber$ and V, respectively. The false positive rate can be expressed as follows:

$$FPR = \frac{V}{Vnumber} \tag{3}$$

D. DETERMINATION OF M VALUE

It is necessary to multiply the data and the matrix B to obtain the value either in watermark embedding or watermark extraction and then calculate the watermark value. There are M elements of '1' in each row and each column in matrix B, which represents that the calculated value of D multiplied by B is related to M data items. Therefore, value of M should be appropriate. If watermark value is associated with more data items, the FPR will be high. If the amount of data associated with watermark value is smaller than FNR will increase. An appropriate value is assigned to M to ensure a balance between FNR and FPR. The test for the M value is shown in figure 3. The ratios of M to N(RMTN) are 35%, 50%, 65% and 80%, for figure 3 (a) to (d), respectively.

Table 3 is obtained by integrating the data in the figure.

Table 3 shows that the sum of FNR and FPR decreases gradually until the ratio of M to N is equal to 0.65. When the ratio of M to N reaches 0.8, the sum of FNR and FPR starts to increase. Therefore, it is appropriate to choose the ratio of M to N as 0.65.

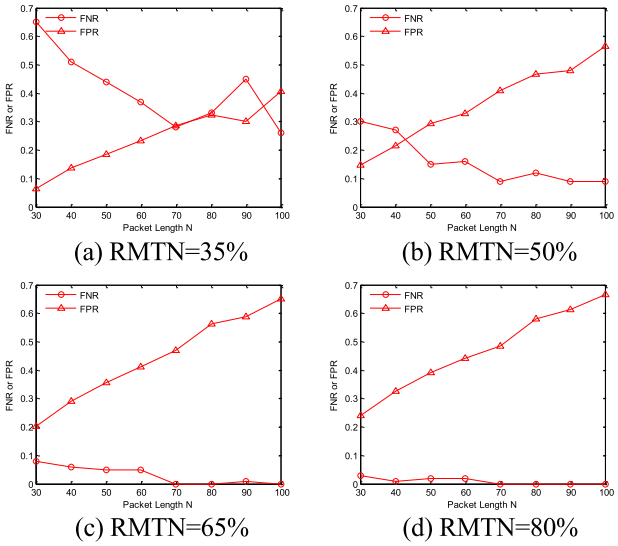


FIGURE 3. Comparison of scheme performance under different M values.

TABLE 3. Results with different M values.

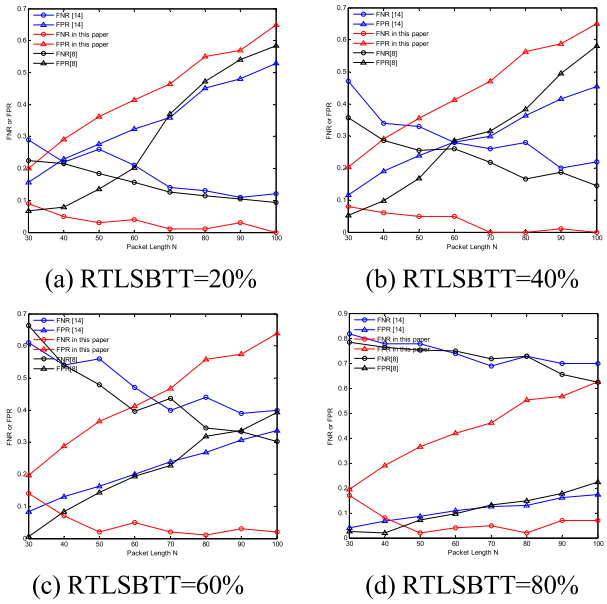
the ratio of M to N	The sum of FNR and FPR
0.3500	5.2255
0.5000	4.1776
0.6500	3.7832
0.8000	3.8212

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section the proposed scheme is demonstrated through MATLAB simulation experiments. The data stream used in the experiment comes from real wireless sensor network in Intel Berkeley Research Laboratory [15], in which the sensors are deployed to collect time stamped information such as humidity, temperature and light intensity. In order to make the experiment simple and clear, a single feature (the temperature) is selected for the experiment. Some obvious errors are removed.

There are usually three ways to tamper with data: insert new data elements, delete data elements and modify data elements. Since the sensor node uses a fixed time interval sampling method, it can be identified whether data is inserted or deleted. Thus, all types of tampering can be seen as modification.

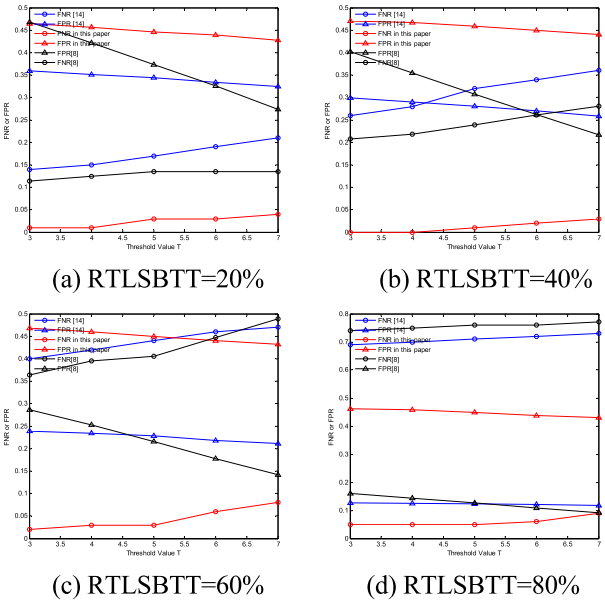
In order to calculate both FNR and FPR, the data stream D is modified at the tamper rate r. Then $D \times r$ data items in the data stream D are tampered. The aim of the experiment in this section is to compare the proposed algorithm with the mainstream independent individual verification algorithms [8], [14]. Since in the case of different tamper rate r, the experimental results of three schemes are almost the same.



(a) RTLSBTT=20% (b) RTLSBTT=40%

(c) RTLSBTT=60% (d) RTLSBTT=80%

FIGURE 4. Comparison of scheme performance under different block length *N*.



(a) RTLSBTT=20% (b) RTLSBTT=40%

(c) RTLSBTT=60% (d) RTLSBTT=80%

FIGURE 5. Comparison of scheme performance under different thresholds *T*.

Therefore, the other three different parameters, the length N of data group, the ratio of M to N and the threshold T are used to test the scheme proposed in this paper. Only the effect of one parameter is tested in each set of experiments. The algorithms proposed in [14] and [8] do not consider the LSB in the process of watermark calculation and embeds the watermarking bit in LSB. When the data is tampered, the LSBs are also likely to be modified, resulting in the instability of the entire algorithm. In figure 4, the ratios of tampered LSBs to the total tampered data (RTLSBTT) are 20%, 40%, 60% and 80%, for figure 4 (a) to (d), respectively.

Figure 4 shows the trends of FNR and FPR of the three schemes with different packet length N. The abscissa is the grouping length of the data stream from 30 to 100, while the ordinate is the value of FNR or FPR. The figure shows that the FNR of the scheme proposed in this paper is better than the schemes proposed in [14] and [8] in all four cases. Although, the FPR has a small increase, in the case of tampering with different proportions of LSB data, the values of FNR and FPR of the algorithm proposed in this paper are relatively stable compared with those of the algorithms proposed in [14] and [8]. For example, when the packet length is 30, and RTLSBTT are from 20% to 80%, the values of FNR and FPR of the proposed algorithm remain around 0.1 and 0.2, respectively. On the contrary, the values of FNR and FPR of the algorithms proposed in [14] and [8] fluctuate greatly. Meanwhile, the sum of FNR and FPR of the algorithm proposed in this paper is less than those of algorithms proposed in [14] and [8].

Figure 5 shows the false negative rate and the false positive rate of the three algorithms under different threshold T. The threshold value T is the key to determine whether a data is tampered or not. When the number of flips in M watermark bits related to the data item s_i is less than the threshold value T, it is considered that the data item is not changed. Otherwise, the data item is changed. It can be seen from the figure that the FNRs of [14] and [8] increase substantially with the increase of the data amount tampered by LSB. However, the trends of FNR and FPR of the algorithm proposed in this paper do not significantly fluctuate in this case. For example, when the threshold value T is equal to 3, more and more data items are tampered with LSB, the values of FNR and FPR of the algorithm proposed in this paper remains around 0.05 and 0.4, respectively. However, the values of FNR and FPR of [14] range from 0.1 to 0.6 and 0.4 to 0.1, respectively. Also, the corresponding values of [8] also range from 0.1 to 0.7 and 0.4 to 0.1, respectively.

In order to further illustrate the superiority of the proposed algorithm, the ratio of M to N is changed keeping the other parameters unchanged. With the change of the ratio of M to N, the FNR and FPR of the algorithm proposed in this paper remain relatively stable in the case of different RTLSBTT. Figure 6 shows the FNR an FPR of the three algorithms under different ratios of M to N. For example, when the ratio of M to N is equal to 0.2, the values of FNR and FPR of the algorithm proposed in this paper are around 0.7 and 0.1, respectively. However, the FNR and FPR of [14] range from 0.8 to 0.9 and 0.1 to 0, respectively. Also, the FNR of [8] range from 0.1 to 0.8. The value of FPR is relatively stable in [8].

It can be seen from Tables 4, 5 and 6 that the sum of FNR and FPR for the algorithm proposed in this paper maintains a relatively stable state with the increase of tampering LSB proportion. As shown in Table 4, the sum of FNR and FPR for the proposed algorithm remains around 4.0, while those rise from about 4.0 to 6.8 in the algorithm proposed in [14]

TABLE 4. Comparison results of two schemes under different group lengths N.

N=30、40、50、60、70、80、90、100									
RTLSBTT	Ref. [14]		Ref. [8]		Proposed algorithm		The sum of FNR and FPR in [14]	The sum of FNR and FPR in [8]	The sum of FNR and FPR in this paper
	Total FNR	Total FPR	Total FNR	Total FPR	Total FNR	Total FPR			
0.2000	1.4800	2.8043	1.2164	2.4526	0.2600	3.5004	4.2843	3.6690	3.7604
0.4000	2.3800	2.3570	1.8771	2.3788	0.2500	3.5332	4.7370	4.2559	3.7832
0.6000	3.8100	1.7284	3.4962	1.7017	0.3600	3.5020	5.5384	5.1979	3.8620
0.8000	5.9400	0.8997	5.7854	0.8992	0.5200	3.4910	6.8397	6.6846	4.0110

TABLE 5. Comparison results of two schemes under different thresholds T.

T=3、4、5、6、7									
RTLSBTT	Ref. [14]		Ref. [8]		Proposed algorithm		The sum of FNR and FPR in [14]	The sum of FNR and FPR in [8]	The sum of FNR and FPR in this paper
	Total FNR	Total FPR	Total FNR	Total FPR	Total FNR	Total FPR			
0.2000	0.8600	1.7137	0.6458	1.8628	0.1200	2.2347	2.5737	2.5086	2.3547
0.4000	1.5600	1.3982	1.2084	1.5423	0.0600	2.2862	2.9582	2.7507	2.3462
0.6000	2.1900	1.1316	2.1042	1.0742	0.2200	2.2500	3.3216	3.1784	2.4700
0.8000	3.5500	0.6153	3.7812	0.6293	0.3000	2.2371	4.1653	4.4105	2.5371

TABLE 6. Comparison results of two schemes under different ratios of M TO N.

Ratio of M to N=0.2、0.3、0.4、0.5、0.6、0.7、0.8、0.9									
RTLSBTT	Ref. [14]		Ref. [8]		Proposed algorithm		The sum of FNR and FPR in [14]	The sum of FNR and FPR in [8]	The sum of FNR and FPR in this paper
	Total FNR	Total FPR	Total FNR	Total FPR	Total FNR	Total FPR			
0.2000	2.9000	2.0384	0.9167	3.7386	1.7000	2.6112	4.9384	4.6553	4.3112
0.4000	3.5700	1.7597	1.6979	3.2411	1.8800	2.5949	5.3297	4.9390	4.4749
0.6000	4.7900	1.1828	2.7604	2.5239	1.8600	2.5718	5.9728	5.2843	4.4318
0.8000	6.3500	0.6701	5.3854	1.3651	1.7900	2.6140	7.0201	6.7505	4.4040

and from about 3.6 to 6.6 in the algorithm proposed in [8], respectively. Under the comparison of different thresholds T shown in Table 5, the sum of FNR and FPR for the algorithm presented in this paper is about 2.3, while those rise from about 2.0 to 4.0 in algorithm proposed in [14] and from about 2.5 to 4.4 in algorithm proposed in [8], respectively. The data in Table 6 shows that the algorithm presented in this paper is more stable than the algorithms proposed in [14] and [8]. The sum of FNR and FPR for the algorithm proposed in this paper remains around 4.0, while those rises from about 4.0 to 7.0 in algorithm proposed in [14] and from about 4.6 to 6.7 in algorithm proposed in [8], respectively.

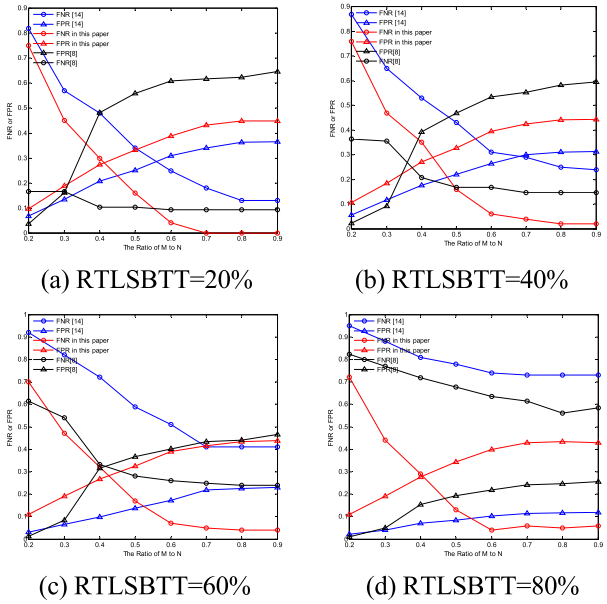
Table 7 provides the comparison of the algorithm proposed in this paper with other wireless sensor network data authentication schemes. The following aspects are mainly compared in the table.

1) Whether the object of verification is a group of data items or a single data item.
2) The object to be embedded in data group.
3) Whether the current data stream is grouped or not.
4) Watermark generation method.
5) The location where watermark is embedded.

In general, Table 7 shows that the proposed scheme is superior to others. The proposed algorithm can verify individual data item, while most of other schemes can only achieve the validation of data group level and are unable to determine whether the individual data item has been tampered with during the data group authentication process. If only a small amount of data items in each data group are modified, the entire data group is discarded and the WSN resources are wasted. However, the watermarking technique based on LSB replacement is not used in the proposed algorithm.

TABLE 7. Comparison of different schemes.

	Zhang[2]	Kamel[1]	Sun[3]	Shi[8]	Dhiman[16]	Hameed[11]	Zhou[6]	Shi[7]	Proposed algorithm
Object of verification	Data group	Data group	Data group	Individual data item	Individual data	Data group	Data group	Data group	Individual data item
Embedded object	Watermark bits	Watermark bits	Watermark bits	Watermark bits	Watermark bits	Watermark bits	Watermark bits	Watermark bits	Spaces
Packet	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Watermark generation method	HASH function	HASH function	HASH function and XOR	HASH function and XOR	HASH function and XOR	XOR	HASH function and XOR	HASH function and XOR	HASH function and XOR
Location of watermark embedding	Current data group	Next data group	Next data group	Current data group	Current data group	Current data group	Current data group	Next data group	Current data group



(a) RTLSBTT=20% (b) RTLSBTT=40%

(c) RTLSBTT=60% (d) RTLSBTT=80%

FIGURE 6. Comparison of scheme performance under different ratio of *M* to *N*.

Because embedding the watermark into LSB of the data item will not only affect the actual value of the data item, but also lose the basis for determining whether the data is modified, especially when the attacker modifies the LSB of the data item. In the proposed algorithm, the space is embedded at the end of data item, which does not change the value of the data item. Therefore, the embedded watermark in the proposed algorithm is more secretive for attackers compared with other algorithms. Finally, the other algorithms generate watermark using current data group and embed watermark into the next data group. However, the watermark generation and embedding in the proposed algorithm are achieved only using the current data group, which may improve the algorithm efficiency.

V. CONCLUSION

This paper proposes a verification scheme for individual data items. The watermark generated by associating other data items can accurately verify the individual data items. Several experiments are conducted on various aspects affecting the FNR and FPR. The obtained results demonstrate the superiority of the proposed algorithm. Furthermore, it is not easy for the attacker to detect the watermark in the proposed scheme. In addition, the embedding of watermark in the proposed scheme does not change the value of the data item, which can reduce the unreliability of watermark contrast caused by tampered data.

REFERENCES

[1] I. Kamel and H. A. Juma, "A lightweight data integrity scheme for sensor networks," *Sensors*, vol. 11, no. 4, pp. 4118–4136, 2011.
[2] D. Zhang, M. Wan, and C. Xu, "False data identification method based on watermarking," *J. Comput.*, vol. 8, no. 10, pp. 2682–2689, 2013.
[3] X. Sun, J. Su, B. Wang, and O. Liu, "Digital watermarking method for data integrity protection in wireless sensor networks," *Int. J. Secur. Appl.*, vol. 7, no. 4, pp. 407–416, 2013.
[4] T. Guan and Y. Chen, "A node clone attack detection scheme based on digital watermark in WSNs," in *Proc. 1st IEEE Int. Conf. Comput. Commun. Internet (ICCCI)*, Oct. 2016, pp. 257–260.
[5] I. Kamel and O. AlKoky, "Semi-fragile watermark for sensor data," *Int. J. Internet Protocol Technol.*, vol. 6, no. 3, p. 156, 2011.
[6] L. Zhou and Z. Zhang, "A secure data transmission scheme for wireless sensor networks based on digital watermarking," in *Proc. 9th Int. Conf. Fuzzy Syst. Knowl. Discovery*, May 2012, pp. 2097–2101.
[7] X. Shi and D. Xiao, "A reversible watermarking authentication scheme for wireless sensor networks," *Inf. Sci.*, vol. 240, pp. 173–183, Aug. 2013.
[8] X. Shi, "A statistical integrity authentication scheme without grouping for streaming data," in *Proc. 9th Int. Conf. P2P, Parallel, Grid, Cloud Internet Comput.*, Nov. 2015, pp. 548–552.
[9] X. Y. Li, Y. J. Zhong, F. B. Liao, and R. Li, "An improved watermarking scheme for secure data aggregation in WSNs," *Appl. Mech. Mater.*, vols. 556–562, pp. 6298–6301, May 2014.
[10] F. Lalem, M. Alshaikh, A. Bounceur, R. Euler, L. Laouamer, L. Nana, and A. Pascu, "Data authenticity and integrity in wireless sensor networks based on a watermarking approach," in *Proc. Int. Florida Artif. Intell. Res. Soc. Conf.*, Mar. 2016, pp. 276–281.
[11] K. Hameed, A. Khan, M. Ahmed, A. G. Reddy, and M. M. Rathore, "Towards a formally verified zero watermarking scheme for data integrity in the Internet of Things based-wireless sensor networks," *Future Gener. Comput. Syst.*, vol. 82, pp. 274–289, May 2018.

[12] W. Baoyi, H. Zepeng, and Z. Shaomin, "Research on WSN secure communication method based on digital watermark for the monitoring of electric transmission lines," *Adv. Comput., Signals Syst.*, vol. 3, no. 2, pp. 8–14, 2019.
[13] B. Wang, W. Kong, W. Li, and N. N. Xiong, "A dual-chaining watermark scheme for data integrity protection in Internet of Things," *CMC-Comput. Mater. Continua*, vol. 58, no. 3, pp. 679–695, 2019.
[14] X. Shi, *Study on the Application of Digital Watermarking in Security of Wireless Sensor Networks*. Chongqing, China: Chongqing Univ., 2015.
[15] *Intel Lab Data*. Accessed: Jul. 20, 2019. [Online]. Available: http://db.lcs.mit.edu/labdata/labdata.html
[16] V. Dhiman and P. Khandnor, "Watermarking schemes for secure data aggregation in wireless sensor networks: A review paper," in *Proc. Int. Conf. Elect., Electron., Optim. Techn. (ICEEOT)*, Mar. 2016, pp. 3093–3098.

YAN XIAO received the B.Eng. degree in information science and technology from Jiujiang University, Jiujiang, China, in 2017.

She is currently pursuing the M.Eng. degree with the College of Information Technology, Jiangxi University of Finance and Economics. Her research interests include computer image and video processing, information security technology, and data mining.

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 and Technology. His research interests include computer networks security, multimedia information security, and digital image processing.