An Energy Conserving Reversible and Irreversible Digital Watermarking Hybrid Scheme for Cluster-based Wireless Sensor Networks

Chu-Fu Wang, An-Ting Wu, Shu-Chien Huang

Department of Computer Science National Pingtung University, Taiwan
cfwang@mail.nptu.edu.tw, antingwu@gmail.com, schuang@mail.nptu.edu.tw

Abstract

Wireless sensor networks (WSNs) are composed of a large number of sensor nodes that are scattered within a monitoring region and in which one or more sinks are responsible for the sensed data collecting. The sensor nodes are equipped with limited battery energy and are generally deployed in unattended environments such as in forests, on volcanoes, etc., to regularly report sensed data to the sink via wireless communication. Therefore, they cannot be recharged when a node depletes its battery energy. To design energy-conserving methods to prolong the network lifetime has become an important issue for WSNs. Nowadays the privacy and accuracy of data transmission are taken seriously; thus this paper aims 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 is proposed to balance the energy consumption and the robustness of the security.

Keywords: Wireless sensor networks, Authentication, Reversible digital watermarking, Irreversible Digital watermarking

1 Introduction

As wireless communication technologies have evolved rapidly, many diverse applications have been proposed for a new era of wireless networks to enhance our daily life. For example, the Vehicular Ad-Hoc Networks (VANETs) can wisely guide drivers to avoid traffic congestion regions, and can also provide multimedia communications for passengers to enhance their comfort and enjoyment within cars. Wireless Sensor Networks (WSNs) can monitor the remote environment status by constantly acquiring sensed data to gain better control of the environment. WSNs [1-2] generally deploy sensor nodes in unattended regions to monitor the status of the environment, such as temperature, humidity, pressure, etc.. Due to the deployed environments generally being harsh, it is not easy to replace the sensor nodes or to recharge their batteries. Thus, how to conserve the usage of battery power of sensor nodes has become an important issue when designing mechanisms in each protocol stack of WSNs. The routing method is one of the factors that significantly affects the performance of energy conservation. Several routing methods have been developed [3-13] to conserve the usage of battery energy of sensor nodes to prolong the network lifetime. Among these works, the LEACH routing protocol [14] organizes the sensor nodes into a two-tiered network architecture called the cluster-based WSN. The LEACH protocol partitions the sensor nodes into a group of clusters. Each cluster will elect one special node called the cluster head to be responsible for message collecting and relaying tasks in the cluster. That is, the sensed data of a sensor node will be firstly sent to its respective cluster head (the intra-cluster communication). The cluster head will then aggregate and relay the sensed data to the sink (the inter-cluster communication) (see Figure 1). Note that the cluster head will be replaced by another healthy sensor node when its battery energy falls below a given threshold after operating for a period of time. In this way, the energy consumption load will be evenly distributed among the sensor nodes within the cluster, and thus the network lifetime of the WSN can be prolonged. In this paper, we regard the cluster-based WSN as the underlying network architecture, and adopt the LEACH routing protocol for message reporting.

Since the sensing nodes are generally deployed in unattended environments, the chance of messages being bugged and tampered with by crackers may increase. Consequently the sensed data will no longer be trusted by the manager of the WSNs, especially for applications in the military or commerce fields. Therefore, the security issue of how to ensure the integrity of the sensed data is another important issue for WSNs. Traditional approaches for data authentication

*Corresponding Author: Chu-Fu Wang; E-mail: cfwang@mail.nptu.edu.tw
DOI: 10.3966/160792642018011901010

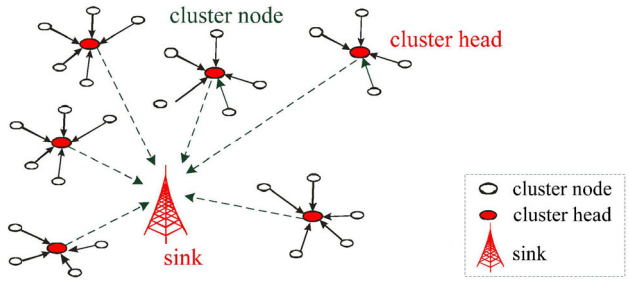

Figure 1. A cluster-based WSN

usually adopt key (secret-key or symmetric-key) encryption schemes using the RC5 encryption algorithm [15] to encrypt (and decrypt) sensed data from the sending side (to the receiving end) [16-18]. However, this approach will cause heavy consumption of battery energy in the sensor nodes, since the data encrypting /decrypting algorithms are complex and the key exchanging process will also greatly contribute to the energy depletion. Applying the digital watermarking techniques [19-22] is one of the promising light-weight schemes compared to the key encryption methods for WSNs when one wants to ensure the integrity of the sensed data while also conserving energy [23-25]. In the existing literature, two types of digital watermarking (reversible and irreversible digital watermarking) are generally applied to the authentication of WSNs. Reversible digital watermarking (also called lossless data embedding) is a technique that enables the encrypted data to be authenticated and then the original sensing data can be restored correctly, while the irreversible digital watermarking cannot recover the embedded data entirely when the sensed data has been attacked. The previous approach is more robust in terms of security than the latter, but obviously it will consume more battery energy.

In this paper, we take both the data authentication issue and the energy saving issue for WSNs into consideration. The considered network architecture is shown in Figure 1 and the LEACH routing protocol is also applied in this cluster-based architecture to conserve battery energy. A Hybrid digital Watermarking Authentication technique (HWA) is proposed to integrate the data authentication process with the routing method under the considered network architecture. The HWA adopts irreversible and reversible data watermarking for the intra-cluster communication and the inter-cluster communication of the WSN, respectively. The main features of the proposed method HWA are described as follows.

(1) HWA provides data authentication and also aims at energy-saving for the cluster-based WSNs;

(2) HWA synthesizes irreversible and reversible digital watermarking for data authentication;

(3) HWA balances the degree of security robustness of data authentication and the energy-saving effect.

The organization of the rest of this paper is as follows. The research survey for applying digital watermarking authentication schemes on WSNs are discussed in Section 2. In Section 3, the proposed method HWA is described in detail. The simulation results and the conclusion are discussed in Section 4 and Section 5, respectively.

2 Research Survey

In this section, the related research works that have applied the digital watermarking authentication method for WSNs are described. Shi and Xiao [23] applied the prediction-error expansion method for data authentication in WSNs, which is designed for embedding/ retrieving the reversible digital watermarking into/from an image. The proposed scheme dynamically groups a chain of data that were constantly sensed and generated by a sensor node. It then computes the respective digital watermarking and embeds it into the next data chain. Note that the proposed scheme regards the sensor nodes of a WSN as a plain architecture rather than a hierarchical structure. The group of sensed data in each sensor node will firstly apply the proposed data encryption method to encrypt the sensed data and then transmit them to the sink along a predetermined routing path. After the sink receives the sensed data, the data decryption method will be invoked to extract the digital watermarking and then to detect the integrity of the received sensed data. Although this method gives a higher degree of security robustness than the irreversible digital watermarking schemes, it will consume more battery energy to perform the encryption and decryption operations compared to the irreversible digital watermarking schemes. In addition, since most of the routing paths pass through the sensor nodes near the sink, these sensor nodes will quickly drain out their battery energy which will shorten the network lifetime of the WSN. Sun *et al.* [24] proposed an authentication scheme for WSNs that applied an irreversible digital watermarking method to protect the data integrity of the sensed data. The proposed method uses redundant space to store the computed digital watermarking and then groups it with the sensed data as a new data unit to transmit to the sink. Similar with the method proposed in [23], the routing scheme also suffers from the drawback of a short network lifetime.

In order to overcome the above drawbacks, Zhou and Zhang [25] adopted the LEACH [14] routing protocol (called the SDTS) to transmit the encrypted sensed data under cluster-based WSN architecture. The proposed authentication scheme applied an irreversible digital watermarking method to compute the digital watermarking with respect to the sensed data and then embedded it into a redundant space. The authentication scheme is applied both in the intra-cluster communication and the inter-cluster communication. Due to the SDTS being a secure data transmission scheme compared with our proposed method, the data encrypting/

decrypting operations will be discussed in detail. In the following, a numerical example will be adopted to demonstrate the operating steps. Given a hash function (h) and a random number generator (g) that is used on both the sending side and the receiving end, the hash function h (and the random number generator g) will generate a 32-bit (and a 16-bit) binary string, respectively. Let S be a 16-bits sensed data (assume that S=1000101100011010). The data encryption operations of the SDTS will firstly calculate a 32-bit hash function value $h_0 = h(S)$ for S. Assume $h_0 = u \cdot l$ = (0010110001000111) · (0001011000100110), where · denotes the string concatenation of two binary strings. Note that, the hash function value h_0 can be partitioned into two parts: the upper part (u=0010110001000111) and the lower part (l=0001011000100110). Then using the random number generator to obtain the key for the watermark embedding (assume g()=1110011010011011).

Now, the sensed data S can be embedded into the hash function value h_0 by replacing its upper part (u) with $S \oplus g()$ =0110110110000001. The resulting data ($S \oplus g()) \cdot l$ (=0110110110000001 · 0001011000100110)) will be the message for transmitting. Note that the lower part of the transmitting message is equal to the lower part (l) of the hash function value (h(S)), which is used for checking whether or not the received data have been tampered with after the receiving end has completely received it.

For the decryption operations of the SDTS, as the data ($S \oplus g()) \cdot l$) are received at the receiving end, since the receiving end has the same random number generator and the hash function, the value g() can be firstly determined. Then the original sensed data S can be easily retrived by applying the exclusive or operation on the upper part of the received data, since $g() \oplus (S \oplus g()) = S$. In addition, in order to determine whether or not the received data have been attacked, one can compare the lower part of the received data (l) with the lower part of the new hash function value $l' = h(S)$. In case of $l = l'$, it can be concluded that the sensed data have not been attacked; On the other hand the sensed data have been tampered with if $l \neq l'$. The above encrypting/ decrypting operations are applied both in the intra-cluster and the inter-cluster communications. In their proposed scheme, due to the length of the redundant space varying and being proportional to the length of the sensed data, the overhead of the battery energy consumption will increase.

3 The Proposed Energy-conserving Hybrid Digital Watermarking Authentication for Cluster-based WSNs

The proposed HWA scheme consists of three steps. They are the intra-cluster authentication (for protecting the sensed data transmission between the sensor nodes and the cluster head), the data aggregation in the cluster head, and the inter-cluster authentication (for protecting the aggregated data transmission between the cluster head and the sink). Note that the operation invoking of the intra-cluster authentication is more frequent, but has a shorter distance of communicating than the inter-cluster authentication. Thus we apply irreversible digital watermarking in the intra-cluster authentication (the Phase I energy conserving intended authentication) to conserve the energy consumption within a cluster, and apply reversible digital watermarking in the inter-cluster authentication (the Phase II security robustness intended authentication) to gain a higher degree of security due to there being a longer range of wireless communication and consequently a higher chance of being attacked than for the intra-cluster authentication.

As shown in Figure 2, the right part of the figure enlarges the illustration of the routing and data authentication for the sensed data reporting, with respect to the area circled by a dashed line on the left of the figure. In this figure, the sensor node u (v) within a cluster will iteratively sense the environment features and collect k (in this figure, k=2) sensed data to form a data group. Then it will embed the respective irreversible digital watermarking into the sensed data group [s_1^u , s_2^u , ..., s_k^u], [s_{k+1}^u , s_{k+2}^u ,..., s_{2k}^u] ... ([s_1^v , s_2^v , ..., s_k^v], [s_{k+1}^v , s_{k+2}^v ,..., s_{2k}^v]...) and send them regularly to the respective cluster head. As the sensed data group arrives at the cluster head, the irreversible digital watermarking will be retracted to check the integrity of the sensed data group by the cluster head. In the case of the sensed data group being verified as not being attacked, the cluster head will aggregate the current received data group from each of the sensor nodes, say, [s_1^u , s_2^u , ..., s_k^u] and [s_1^v , s_2^v , ..., s_k^v] (and [s_{k+1}^u , s_{k+2}^u ,..., s_{2k}^u] and [s_{k+1}^v , s_{k+2}^v ,..., s_{2k}^v], …, etc.) to be [s_1 , s_2 , ..., s_k] ([s_{k+1} , s_{k+2} , ..., s_{2k}], ..., etc.). Then, the aggregated data will be encrypted to embed the reversible digital watermarking and then relay it to the sink. Finally, the sink will check the integrity of the received aggregated data. In the case of the data being verified as not being attacked, the aggregated data will be passed to the upper layer; otherwise, it will simply be discarded. A detailed data flow chart of the HWA is shown in Figure 3. The operations in each phase of the authentication are as follows.

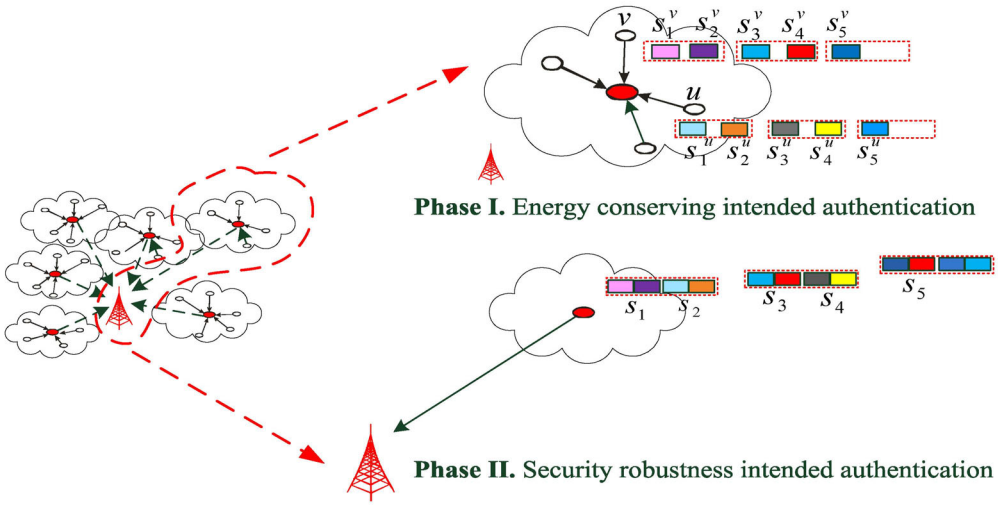

Figure 2. An illustration of the operations of the HWA

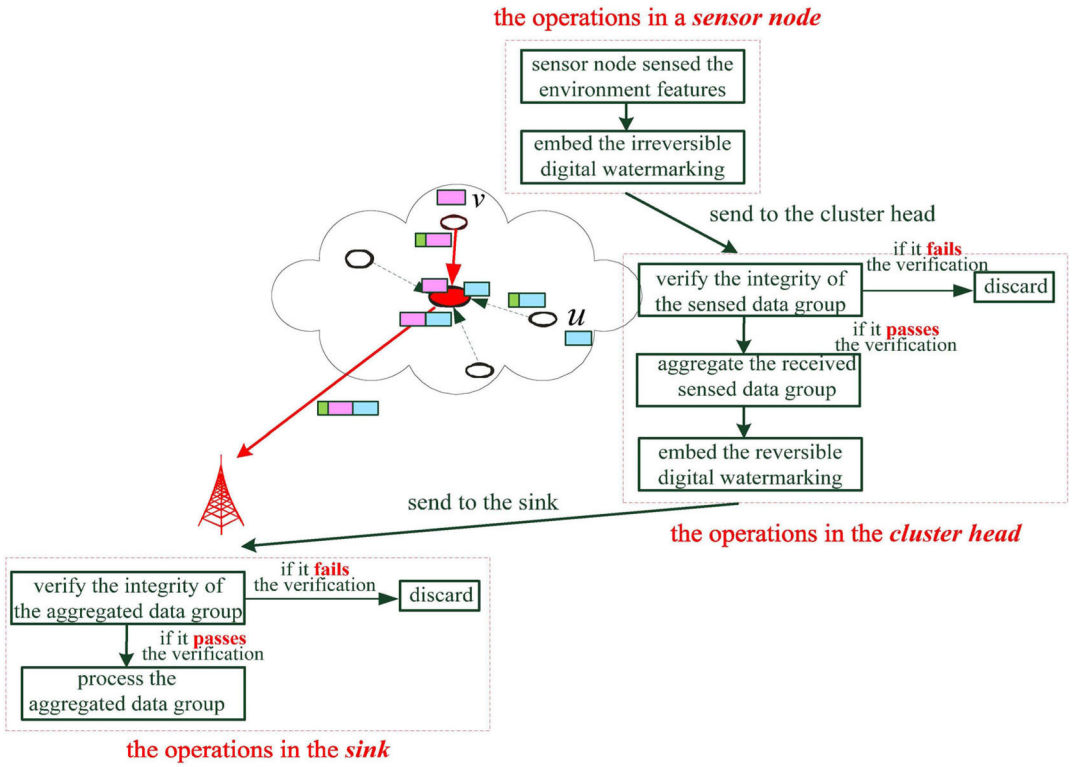

Figure 3. The data processing flowchart of the HWA

3.1 The Phase I Energy Conserving Intended Authentication

This phase of authentication adopts an irreversible digital watermarking scheme to protect the data communication between the sensor nodes within a cluster and their respective cluster head. The operations of the authentication are as follows.

Digital watermarking embedding. Let $s_i^v, i \in \mathrm{N}$ denotes the sequence of sensed data that is generated by sensor node v, and k denotes the size of a sensed data group. Let h be the hash function for generating the digital watermarking w. For each sensed data group $[s_{a\cdot k+1}^v, s_{a\cdot k+2}^v, \ldots, s_{(a+1)\cdot k}^v]$, where $a = 0, 1, 2, \ldots$, the hash function values $h(s_{a\cdot k+1}^v)$, $h(s_{a\cdot k+2}^v)$, …, $h(s_{(a+1)\cdot k}^v)$ are firstly determined (see the upper right of Figure 4 (a=0 in this figure)). The digital watermarking (w) is equal to the resulting value that applies the exclusive or operation on all of the hash function values; that is, $w = h(s_{a\cdot k+1}^v) \oplus h(s_{a\cdot k+2}^v) \oplus \ldots \oplus h(s_{(a+1)\cdot k}^v)$. We then use a redundant area for embedding the digital watermarking w into the string that is concatenated by all of the sensed data $s_{a\cdot k+1}^v$, $s_{a\cdot k+2}^v$, …, $s_{(a+1)\cdot k}^v$. The resulting string is denoted by D; that is,

$D = w \cdot (s^v_{a \cdot k+1} \cdot s^v_{a \cdot k+2} \cdot \ldots \cdot s^v_{(a+1) \cdot k})$. Finally, the sensor node will send the resulting data string D to the cluster head.

Digital watermarking extraction and verification. Let $\hat{D} = \hat{w} \cdot (\hat{s}^v_{a \cdot k+1} \cdot \hat{s}^v_{a \cdot k+2} \cdot \ldots \cdot \hat{s}^v_{(a+1) \cdot k})$ denote the received sensed data group in the cluster head that was sent from sensor node v. The verification steps are quite simple and similar to the operations of the digital watermarking embedding. Note that the cluster head has the same hash function that is used in the same cluster. At first, the cluster head extracts the received digital watermarking $\hat{w}$ and the current received sensed data group $[\hat{s}^v_{a \cdot k+1}, \hat{s}^v_{a \cdot k+2}, \ldots, \hat{s}^v_{(a+1) \cdot k}]$. For the received sensed data group, the cluster head recomputed the digital watermarking by $w = h(\hat{s}^v_{a \cdot k+1}) \oplus h(\hat{s}^v_{a \cdot k+2}) \oplus \ldots \oplus h(\hat{s}^v_{(a+1) \cdot k})$. It then compares the received digital watermarking $\hat{w}$ with the recomputed digital watermarking w (see the lower left of the Figure 4). In the case of the result of the verification showing that the values w and $\hat{w}$ are not equal, then the received sensed data group is claimed to be attacked and we will simply discard these data; otherwise, the received sensed data group will be kept for later processing.

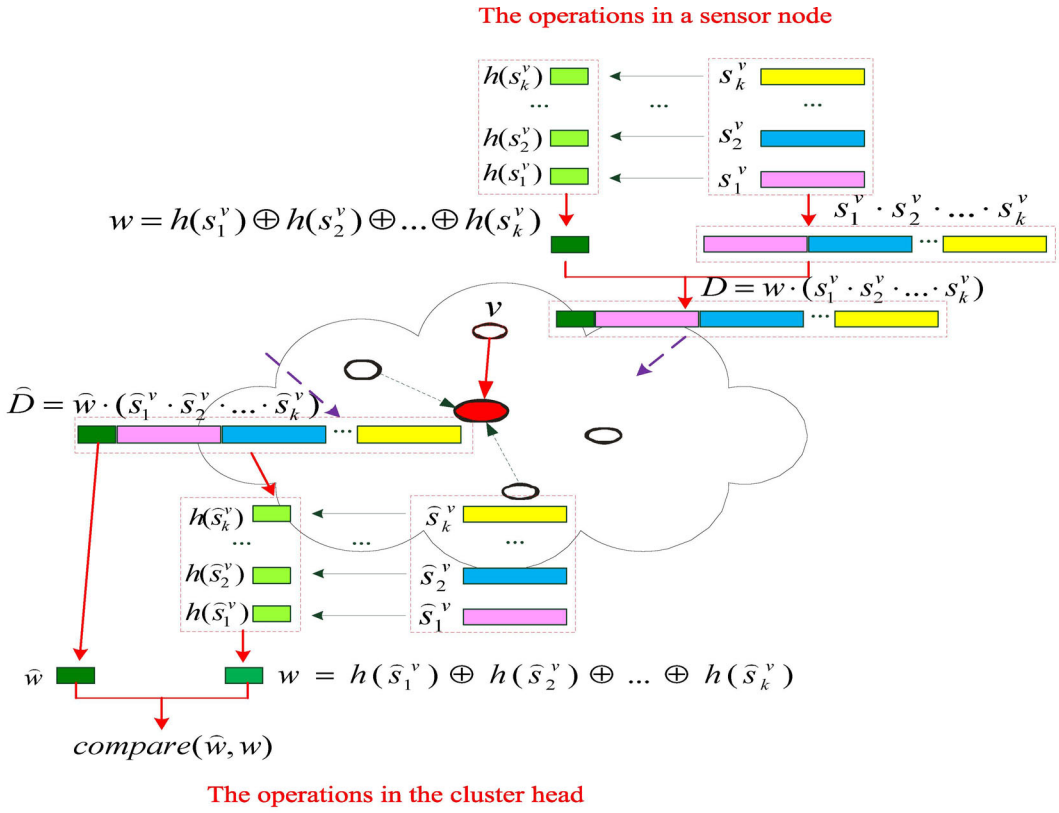

Figure 4. An illustration of the operations of phase I authentication in the HWA

3.2 The Phase II Security Robustness Intended Authentication

After a cluster head receives the authentic sensed data group sent by the sensor nodes, the cluster head u^* will aggregate them constantly to form a new message $a_i^{u^*}, i \in \mathrm{N}$ ready for manipulating by the phase II authentication. This phase of authentication adopts a reversible digital watermarking scheme [23] called the prediction-error expansion to protect the data communication between the cluster head and the sink. The prediction-error expansion has been proven to be notable for multimedia fields. The phase II authentication uses a hash function h and a message group parameter m to dynamically group the aggregated messages and also to determine the digital watermarking. The dynamic group operations are as follows. Let $h(a_i^{u^*})$ be the hash function values with respect to the authenticated aggregated message $a_i^{u^*}, i \in \mathrm{N}$ in phase I of the cluster head u^*. In the case of the hash function value of an aggregated message $a_i^{u^*}$ being congruent to 0 modulo m; that is, $h(a_i^{u^*}) \bmod m \equiv 0$, then this aggregated message is called a *synchronization point*. The aggregated messages between two synchronization points plus the second synchronization point form a message group.

For example, assume the aggregated message $a_i^{u^*}$, $a_j^{u^*}$, $a_k^{u^*}$, $a_l^{u^*}$ shown in Figure 5 are synchronization

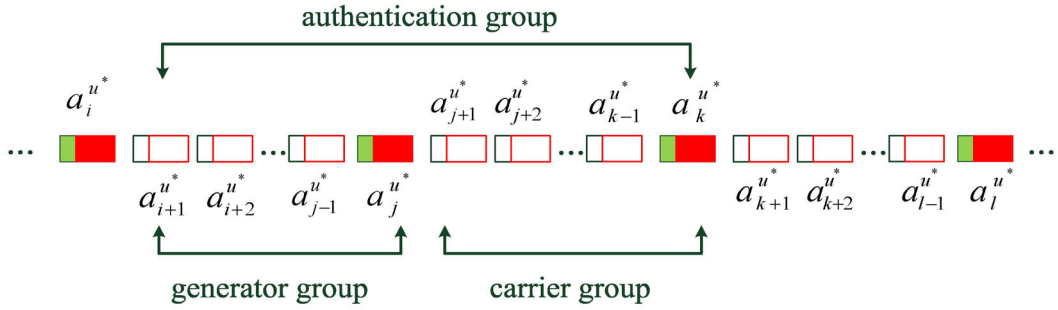

Figure 5. Aggregated message grouping for the phase II authentication of the HWA [23]

points (that is, $h(a_i^{u^*}) \bmod m \equiv 0$, $h(a_j^{u^*}) \bmod m \equiv 0$, and $h(a_k^{u^*}) \bmod m \equiv 0$). Then, the aggregated messages $[a_{i+1}^{u^*}, a_{i+2}^{u^*}, ..., a_{j-1}^{u^*}, a_j^{u^*}]$, $[a_{j+1}^{u^*}, a_{j+2}^{u^*}, ..., a_{k-1}^{u^*}, a_k^{u^*}]$, and $[a_{k+1}^{u^*}, a_{k+2}^{u^*}, ..., a_{l-1}^{u^*}, a_l^{u^*}]$ are three aggregated message groups in Figure 5. And two adjacent aggregated message groups will composite to form an authentication group in the phase II authentication. The previous group of the authentication group is called a generator group and the second group is called a carrier group. The aggregated message in the generator group will firstly be used to determine a digital watermarking and then embed it into the carrier group. Similarly, the carrier group will become the generator group of the next adjacent aggregated message group. For example, the aggregated message group $[a_{i+1}^{u^*}, a_{i+2}^{u^*}, ..., a_{j-1}^{u^*}, a_j^{u^*}]$ and $[a_{j+1}^{u^*}, a_{j+2}^{u^*}, ..., a_{k-1}^{u^*}, a_k^{u^*}]$ ($[a_{j+1}^{u^*}, a_{j+2}^{u^*}, ..., a_{k-1}^{u^*}, a_k^{u^*}]$ and $[a_{k+1}^{u^*}, a_{k+2}^{u^*}, ..., a_{l-1}^{u^*}, a_l^{u^*}]$) forms an authentication group. $[a_{i+1}^{u^*}, a_{i+2}^{u^*}, ..., a_{j-1}^{u^*}, a_j^{u^*}]$ (and $[a_{j+1}^{u^*}, a_{j+2}^{u^*}, ..., a_{k-1}^{u^*}, a_k^{u^*}]$) is the generator group and the $[a_{j+1}^{u^*}, a_{j+2}^{u^*}, ..., a_{k-1}^{u^*}, a_k^{u^*}]$ (and $[a_{k+1}^{u^*}, a_{k+2}^{u^*}, ..., a_{l-1}^{u^*}, a_l^{u^*}]$) is the carrier group. In the following, we will use a numerical example to illustrate how the phase II authentication scheme works.

Digital watermarking generating. As shown in Figure 5, let $a_i^{u^*}$, $a_j^{u^*}$, and $a_k^{u^*}$ be three adjacent synchronization points; and let $[a_{i+1}^{u^*}, a_{i+2}^{u^*}, ..., a_{j-1}^{u^*}, a_j^{u^*}]$ be the generator group and $[a_{j+1}^{u^*}, a_{j+2}^{u^*}, ..., a_{k-1}^{u^*}, a_k^{u^*}]$ be the carrier group. Thus the generator group will be used to generate the digital watermarking for embedding into the carrier group. For the generator group $[a_{i+1}^{u^*}, a_{i+2}^{u^*}, ..., a_{j-1}^{u^*}, a_j^{u^*}]$, the respective hash function values $[h(a_{i+1}^{u^*}), h(a_{i+2}^{u^*}), ..., h(a_{j-1}^{u^*}), h(a_j^{u^*})]$ can be firstly determined. Notice that, since $a_j^{u^*}$ is a synchronization point, thus, $h(a_j^{u^*}) \bmod m \equiv 0$ the digital watermarking can be obtained by $w = h(a_{i+1}^{u^*}) \oplus h(a_{i+2}^{u^*}) \oplus \quad ... \oplus h(a_{j-1}^{u^*}) \oplus h(a_j^{u^*})$. Then convert the decimal digital watermarking w into the equivalent binary bits string of length n; for example, $w = b_n b_{n-1} b_{n-2} ... b_2 b_1$. Then, for each data bit b_t of w, it will be embedded into the respective aggregated message $a_{j+t}^{u^*}$ of the carrier group. Note that, if the length n of binary string w is less than the number of aggregated messages of the carrier group, then the data bit for embedding will be repeated to start from the first data bit of w.

Numerical example:

Assume the aggregated message ($a_t^{u^*}$) of cluster head u^* is [...,**2**,8,9,0,1,8,**2**,3,7,4,5,**6**,...] and the grouping parameter m=4. Assume the hash function values $h(a_t^{u^*})$ with respect to the aggregated messages are [...,**4**,6,7,5,3,6,**4**,1,2,3,9,**8**,...]. Since the hash function values (4 and 8) of the aggregated message 2 and 6 are congruent with 0 modulo m=4, the aggregated data 2, 2, and 6 are the synchronization points. The data group [8,9,0,1,8,**2**] becomes the generator group and [3,7,4,5,**6**] becomes the carrier group. The digital watermarking w can be obtained as follows, $w = h(8) \oplus h(9) \oplus h(0) \oplus h(1) \oplus h(8) \oplus h(2) = 6 \oplus 7 \oplus 5 \oplus 3 \oplus 6 \oplus 4 = 0110 \oplus 0111 \oplus 0101 \oplus 0011 \oplus 0110 \oplus 0100 = 0101$.

Digital watermarking embedding. Let the carrier group be $[a_{j+1}^{u^*}, a_{j+2}^{u^*}, ..., a_{k-1}^{u^*}, a_k^{u^*}]$ and the digital watermarking for embedding be $w = b_n b_{n-1} b_{n-2} ... b_2 b_1$. The prediction-error expansion method embeds each digital watermarking bit b_1, b_2, ... b_{n-1}, b_n, b_1, b_2, ... b_{n-1}, b_n, ... into each aggregated message value $a_{j+1}^{u^*}$, $a_{j+2}^{u^*}$, ..., $a_{k-1}^{u^*}$, and $a_k^{u^*}$, respectively. Let the resulting aggregated message group after embedding be $[\hat{a}_{j+1}^{u^*}, \hat{a}_{j+2}^{u^*}, ..., \hat{a}_{k-1}^{u^*}, \hat{a}_k^{u^*}]$ and then the cluster head will send this message to the sink for authenticating. Detailed embedding operations are stated as follows. Let the current aggregated message value for embedding be $a_t^{u^*}$ and the respective digital watermarking bit for embedding be b_t. At first, the prediction-error expansion method will determine the respective predicted value s_t' and the prediction-error p_t using the following equations, $s_t' = \lfloor (a_{t-2}^{u^*} + a_{t-1}^{u^*})/2 \rfloor$, where

$a_{t-1}^{u^*}$ and $a_{t-2}^{u^*}$ denotes the two previous aggregated message values with respect to the current aggregate message value $a_t^{u^*}$ **(1)**

$$p_t = a_t^{u^*} - s_t' \quad \textbf{(2)}$$

Then the resulting encrypted value $\hat{a}_t^{u^*}$ with respect to $a_t^{u^*}$ can be determined as follows,

$$\hat{a}_t^{u^*} = s_t' + 2 \cdot p_t + b_t \quad \textbf{(3)}$$

Numerical example:

Continue the above numerical example. Note that, the digital watermarking w is 01010101... ($b_4 b_3 b_2 b_1 b_4 b_3 b_2 b_1$...) and the carrier group is [3,7,4,5,**6**]. Now, assume the current manipulating aggregated message value $a_t^{u^*}$ is 6, and the respective current digital watermarking bit for embedding b_1 is 1 (since the embedding sequence is b_1, b_2, b_3, b_4, and b_1...). The previous two aggregated message values $a_{t-1}^{u^*} = 5$ and $a_{t-2}^{u^*} = 4$. The predicted value can then be determined by $s_t' = \lfloor (4+5)/2 \rfloor = 4$ and the prediction error $p_t = 6 - 4 = 2$. The encrypted aggregated message value $\hat{a}_t^{u^*}$ is equal to $s_t' + 2 \cdot p_t + b_t = 4 + 2 \cdot 2 + 1 = 9$. Continue the above operations; all of the encrypted aggregated message values in the carrier group can then be determined.

Digital watermarking extraction and verification. As the sink received an authentication group $[\hat{a}_{i+1}^{u^*}, \hat{a}_{i+2}^{u^*}, ..., \hat{a}_{j-1}^{u^*}, \hat{a}_j^{u^*}]$ and $[\hat{a}_{j+1}^{u^*}, \hat{a}_{j+2}^{u^*}, ..., \hat{a}_{k-1}^{u^*}, \hat{a}_k^{u^*}]$, where $[\hat{a}_{i+1}^{u^*}, \hat{a}_{i+2}^{u^*}, ..., \hat{a}_{j-1}^{u^*}, \hat{a}_j^{u^*}]$ is the generator group and $[\hat{a}_{j+1}^{u^*}, \hat{a}_{j+2}^{u^*}, ..., \hat{a}_{k-1}^{u^*}, \hat{a}_k^{u^*}]$ is the carrier group. The prediction-error expansion method will apply the above embedding process to obtain a new digital watermarking w. Then apply a digital watermarking extraction process, which will be described later, to extract the original embedded digital watermarking $\hat{w}$. Obviously, if the two values are equal (that is, $w = \hat{w}$), then the data in the generator group are claimed to pass the verification; otherwise, they were attacked during transmission. The digital watermarking extraction process is stated as follows.

Let the current manipulating received aggregated message value be $\hat{a}_t^{u^*}$. Since the embedded digital watermarking is generated by the previous group, the embedded digital watermarking has also been determined at this time. Moreover, the previous two aggregated message values $a_{t-1}^{u^*}$ and $a_{t-2}^{u^*}$ before embedding have also been correctly revealed. Assume the respective embedded digital watermarking bit for value $\hat{a}_t^{u^*}$ is $\hat{b}_t$. The only values waiting to be determined are the embedded digital watermarking bit $\hat{b}_t$ and the original message value $\hat{a}_t^{u^*}$. Similar to the embedding process and to reverse the operations (one can refer to the prediction-error expansion method for detailed), the respective predicted value s_t' can be obtained by Equations (1) ($s_t' = \lfloor (a_{t-2}^{u^*} + a_{t-1}^{u^*})/2 \rfloor$) and let the new prediction error p_t' be $\hat{a}_t^{u^*} - s_t'$). The embedded digital watermarking bit $\hat{b}_t$ can be obtained by retracting the least significant bit of p_t'; that is $\hat{b}_t = LSB(p_t')$. The original aggregated message value $a_t^{u^*}$ can be determined by $a_t^{u^*} = \hat{a}_t^{u^*} - \lfloor p_t'/2 \rfloor - \hat{b}_t$.

Numerical example:

Continue the above numerical example. Note that the digital watermarking w is 01010101... ($b_4 b_3 b_2 b_1 b_4 b_3 b_2 b_1$...). Assume the current manipulating received aggregated message data value $\hat{a}_t^{u^*}$ is 9. And the two previous aggregated message values have been revealed and we have that $a_{t-1}^{u^*} = 5$ and $a_{t-2}^{u^*} = 4$. Now, we want to extract the respective embedded digital watermarking bit $\hat{b}_t$ and the original aggregated message value $a_t^{u^*}$ (as shown in the numerical example of the digital watermarking embedding, $\hat{b}_t = 1$ and $a_t^{u^*} = 6$). Firstly, the respective predicted value and the new prediction error can be obtained by $s_t' = \lfloor (a_{t-2}^{u^*} + a_{t-1}^{u^*})/2 \rfloor = \lfloor (4+5)/2 \rfloor = 4$ and $p_t' = \hat{a}_t^{u^*} - s_t' = 9 - 4 = 5$. Since the binary representation of $p_t' = 5 = 0101$, the embedded digital watermarking bit $\hat{b}_t = LSB(p_t') = 1$, and the original aggregated message value $\hat{a}_t^{u^*}$ can be recovered by $a_t^{u^*} = \hat{a}_t^{u^*} - \lfloor p_t'/2 \rfloor - \hat{b}_t = 9 - \lfloor 5/2 \rfloor - 1 = 6$, which matches the values shown in the previous numerical example.

4 The Simulation Results

The research used the C++ programming language to build the simulation environment. The sensing region was set to be a 300m*300m square area and the sink was located at coordinate (0,0). The number of sensor nodes is 200 and they were randomly deployed within the sensing region. The simulation adopts a cluster-based WSN and uses the LEACH protocol as the message routing mechanism (the probability of a sensor node to act as a cluster head is set to be 5%). For the simulations in the Phase I authentication, a sensor node will collect 50 sensed data values and form a sensed data group to send to the cluster head. The number of transmission rounds performed in each comparison is set to be 6,000 rounds. For each performance evaluation, 50 simulations have been performed to take their average results. The details of the environment parameter settings are shown in Table 1.

Table 1. The simulation parameter settings

Parameter	Value	Parameter	Value
Sensing area	300m*300m	Number of sensor nodes	200
Cluster head probability	5%	The size of a sensed data group in sensor nodes per round	50
Number of simulation rounds	6000	Attacking rate	0.5%~5%

The performance evaluation parameters in the simulations include the network lifetime, the delivery ratio, the data false positive rate, and the data false negative rate. The definitions of these comparison parameters are as follows.

The WSN's *Network Lifetime (LT).* The network lifetime is defined to be the number of sensed data transmission rounds that can be successfully performed before the first sensor node drains out its battery energy. Note that, since the amount of energy consumed for instruction execution is much less than for the data transmission (generally, the energy consumed for executing 3,000 instructions is equal to the energy consumed for transmitting 1 bit data to 100m), thus we ignore the energy consumed for data manipulating in our simulations.

The data *Delivery Ratio (DR).* The data delivery ratio is defined as the number of messages successfully received by the sink divided by the total number of messages sent from the sensor nodes. That is,

$$DR = \frac{\text{Number of successfully received messages}}{\text{Total number of messages sent}}$$

The data *False Positive Rate (FPR).* The data false positive rate is defined as the number of messages that fail to pass the authentication, but have not been attacked, divided by the total number of messages sent from the sensor nodes. That is,

$$FPR = \frac{\text{Number of message fails to pass the authentications, but have not been attacked}}{\text{Total number of messages sent}}$$

The data *False Negative Rate (FNR).* The data false negative rate is defined as the number of messages that passed the authentication, but which have been attacked, divided by the total number of messages sent from the sensor nodes. That is,

$$FNR = \frac{\text{Number of messages which passesed the authentications, but which have been attacked}}{\text{Total number of messages sent}}$$

Based on the above definitions, a larger network lifetime (and a delivery ratio) stands for the respective authentication scheme being more energy-conserving (and more reliable) than the other one. Similarly, a lower data false positive rate (and data false negative rate) with respect to an authentication scheme stands for it being more security robust than the other one. For the compared method, since the authentication scheme proposed by Zhou and Zhang [25] (called the SDTS) has the same considered environment architecture (the cluster-based WSN) as our proposed method, it was implemented to be the comparison method with our proposed scheme HWA. The simulation results are discussed as follows.

The network lifetime comparison results are shown in Figure 6. In this figure, the proposed approach, HWA, outperformed the SDTS method. The upper and lower simulation result range is given under a confidence interval of 0.9. The network lifetime for the HWA can endure around 7,000 rounds of message transmissions for any attacking rate; however, the SDTS can only endure around 2,300 rounds of message transmissions. This is because the message length of the SDTS is much longer than that of the HWA; consequently, the amount of energy consumed per round in the SDTS scheme is larger than that consumed by HWA. Figure 7 shows the performance results for the message delivery ratio comparisons. As the attacking rate increases, the message delivery ratio will decrease accordingly for both schemes. The performance curve of the proposed HWA in this figure are all greater than 97% for any attacking rate and are better than SDTS.

The data false positive rate (and the data false negative rate) comparison results are shown in Figure 8 (and Figure 9), respectively. As shown in these figures, the data false positive rate and the data false negative rate for the proposed scheme, HWA, will slightly increase as the attacking rate increases, but the performance results are all below 0.01. The results also show that the SDTS performed better than HWA in terms of the data false positive rate comparisons and the data false negative rate comparisons. However, the network lifetime performance and the data delivery ratio of the SDTS are worse than that of the HWA.

5 Conclusion

Nowadays WSN applications are common in our daily life, including human health monitoring, military applications, or even integrating cloud computing to manipulate the huge amount of sensed data for more important applications. However, energy conservation and the security issues are both important, but there is a tradeoff in the WSNs design. In this paper, a lightweight hybrid authentication scheme that synthesizes reversible and irreversible digital watermarking techniques for the cluster-based WSN is proposed to enhance the security of the sensed data reporting and also to conserve the battery usage of the sensor nodes. The simulation results show that the proposed scheme outperformed

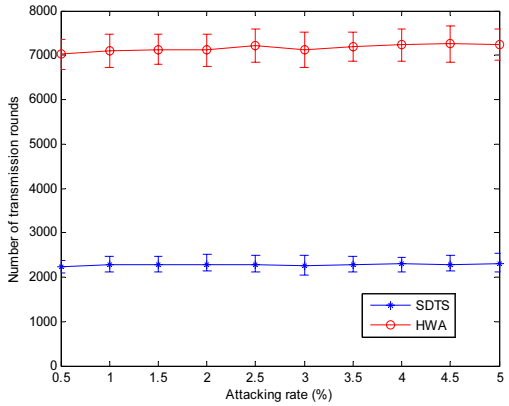

Figure 6. Network lifetime comparison results

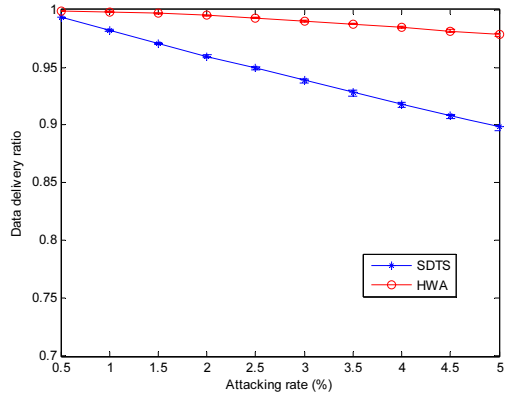

Figure 7. Data delivery ratio comparison results

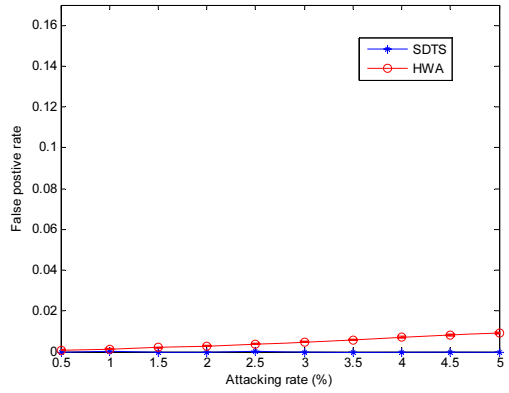

Figure 8. Data false positive rate comparison results

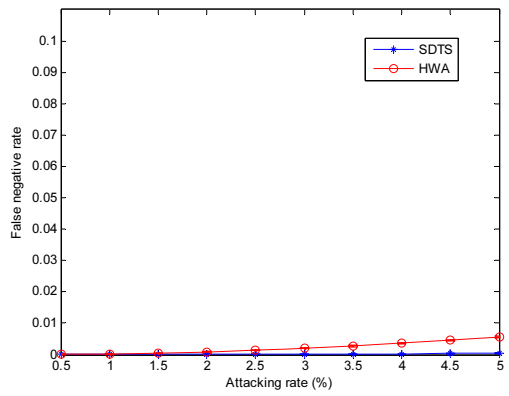

Figure 9. Data false negative rate comparison results

the compared method in network lifetime and the delivery ratio comparisons. The data false positive rate comparisons and the data false negative rate comparisons of the proposed method only slightly less than the performance results of the compared method.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, A Survey on Sensor Networks, *IEEE Communications Magazine*, Vol. 40, No. 8, pp. 102-114, August, 2002.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless Sensor Network: A Survey, *Computer Networks*, Vol. 38, No. 4, pp. 393-422, March, 2002.

[3] J. N. Al-Karaki, A. E. Kamal, Routing Techniques in Wireless Sensor Networks: A Survey, *IEEE Wireless Communications Magazine*, Vol. 11, No. 6, pp. 6-28, December, 2004.

[4] A. Manjeshwar, Q. Zeng, D. P. Agrawal, An Analytical Model for Information Retrieval in Wireless Sensor Networks Using Enhanced APTEEN Protocol, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 13, No. 12, pp. 1290-1302, December, 2002.

[5] X. Hong, M. Gerla, H. Wang, L. Clare, Load Balanced, Energy Aware Communications for Mars Sensor Networks, *IEEE Aerospace Conference, Vol. 3*, Big Sky, MT, 2002, pp. 1109-1115.

[6] S. C. Huang, R. H. Jan, Energy-aware, Load Balanced Routing Schemes for Sensor Networks, *10th International Conference on Parallel and Distributed Systems*, Newport Beach, CA, 2004, pp. 419-425.

[7] R. C. Shah, J. M. Rabaey, Energy Aware Routing for Low Energy Ad Hoc Sensor Networks, *IEEE Wireless Communications and Networking Conference*, Orlando, FL, 2002, pp. 350-355.

[8] H. R. Karkvandi, E. Pecht, O. Yadid-Pecht, Effective Lifetime-aware Routing in Wireless Sensor Networks, *IEEE Sensors Journal*, Vol. 11, No. 12, pp. 3359-3367, December, 2011.

[9] I. S. AlShawi, L. Yan, W. Pan, B. Luo, Lifetime Enhancement in Wireless Sensor Networks Using Fuzzy Approach and A-star Algorithm, *IEEE Sensors Journal*, Vol. 12, No. 10, pp. 3010-3018, October, 2012.

[10] S. S. Wang, Z. P. Chen, LCM: A Link-aware Clustering Mechanism for Energy-efficient Routing in Wireless Sensor Networks, *IEEE Sensors Journal*, Vol. 13, No. 2, pp. 728-736, February, 2013.

[11] K. Kalpakis, K. Dasgupta, P. Namjoshi, Efficient Algorithms for Maximum Lifetime Data Gathering and Aggregation in Wireless Sensor Networks, *Computer Networks*, Vol. 42, No. 6, pp. 697-716, August, 2013.

[12] U. Monaco, F. Cuomo, T. Melodia, F. Ricciato, M. Borghini, Understanding Optimal Data Gathering in the Energy and Latency Domains of a Wireless Sensor Network, *Computer Networks*, Vol. 50, No. 18, pp. 3564-3584, December, 2006.

[13] W. R. Heinzelman, J. Kulik, H. Balakrishnan, Adaptive

Protocols for Information Dissemination in Wireless Sensor Networks, *5th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Seattle, Washington, 1999, pp. 174-185.

[14] M. J. Handy, M. Haase, D. Timmermann, Low Energy Adaptive Clustering Hierarchy with Deterministic Cluster-head Selection, *IEEE 4th International Workshop of Mobile and Wireless Communications Network*, Stockholm, Sweden, 2002, pp. 368-372.

[15] R. Rivest, The RC5 Encryption Algorithm, *Fast Software Encryption of Lecture Notes in Computer Science*, B. Preneel, ed., pp. 86-96, Springer-Verlag, 1995.

[16] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, D. E. Culler, SPINS: Security Protocols for Sensor Networks, *Wireless Networks*, Vol. 8, No. 5, pp. 521-534, September, 2002.

[17] K. K. Suraj, K. R. Radhika, A Novel Symmetric Key Encryption Algorithm Based on RC5 in Wireless Sensor Network, *International Journal of Emerging Technology and Advanced Engineering*, Vol. 3, No. 6, pp. 373-376, June, 2013.

[18] A. Perrig, J. Stankovic, D. Wagner, Security in Wireless Sensor Networks, *Communications of the ACM*, Vol. 47, No. 6, pp. 53-57, June, 2004.

[19] C. T. Hsu, J. L. Wu, Hidden Digital Watermarks in Image, *IEEE Transactions on Consumer Electronics*, Vol. 44, No. 1, pp. 206-216, 1998.

[20] C. S. Lu, S. K. Huang, C. J. Sze, H. Y. M. Liao, Cocktail Watermarking for Digital Image Protection, *IEEE Transactions on Multimedia*, Vol. 2, No. 4, pp. 209-224, December, 2000.

[21] S. Bounkong, B. Toch, D. Saad, D. Lowe, ICA for Watermarking Digital Images, *Journal of Machine Learning Research*, Vol. 4, No. 7-8, pp. 1471-1498, October-Novemver, 2003.

[22] R. G. van Schyndel, A. Z. Tirkel, C. F. Osbome, A Digital Watermark, *IEEE International Conference Image Processing (ICIP'94,)*, Austin, TX, 1994, pp. 86-90.

[23] X. Shi, D. Xiao, A Reversible Watermarking Authentication Scheme for Wireless Sensor Networks, *Information Sciences*, Vol. 240, pp. 173-183, August, 2013.

[24] X. Sun, J. Su, B. Wang, Q. Liu, Digital Watermarking Method for Data Integrity Protection in Wireless Sensor Networks, *International Journal of Security and Its Applications*, Vol. 7, No. 4, pp. 407-416, July, 2013.

[25] L. Zhou, Z. Zhang, A Secure Data Transmission Scheme for Wireless Sensor Networks Based on Digital Watermarking, *the 9th International conference on Fuzzy Systems and Knowledge Discovery (FSKD'12)*, Sichuan, China, 2012, pp. 2097-2101.

Biographies

Chu-Fu Wang received the B.S. degree in Applied Mathematics from National Cheng Kong University, and the M.S. and Ph.D. degrees in Computer and Information Science from National Chiao Tung University, Taiwan, in 1993, 1995, and 2001, respectively. He joined the Department of Computer Science, National Pingtung University, in 2002, where he is currently a Professor. His research interests include multicast distribution, mobile computing, and network optimization.

An-Ting Wu received the M.S. degree in computer science from National Pingtung University, Taiwan, in 2015. Her research interests include network security and mobile computing.

Shu-Chien Huang received the Ph.D. degree in Computer Science and Information Engineering from National Cheng Kung University, Taiwan, Republic of China, in 1999. He is currently an associate Professor at the Department of Computer Science in National Pingtung University. His researching interests include image processing, pattern recognition, and swarm and evolutionary computation.