Lightweight load-balanced and authentication scheme for a cluster-based wireless sensor network

Jiliang Zhou¹ and Ziqiang Lin²

Abstract
Clustering technology is one of the crucial technologies to prolong the lifetime in wireless sensor networks. However, most cluster schemes choose cluster head randomly to send data without considering load balancing and security. In addition, some cluster heads in the highly active area may be overloaded, while others in the low active area may be overloaded, which may easily lead to extreme imbalance in task allocation. Our research on relevant literature shows that the existing authentication schemes do not fully consider the load balancing of cluster heads, while the load balancing schemes ignore the security authentication of cluster heads. Therefore, this article effectively combines load balancing and security verification, and proposes a lightweight load balancing and verification scheme (secure load and energy balancing) based on clustered wireless sensor networks. Secure load and energy balancing implements cluster head’s authentication and confidentiality and integrity of all messages in load balancing. This scheme not only effectively maintains the energy balance of the whole network but also successfully improves the security overhead, thus prolonging the network lifetime. The simulation results show that compared with other similar schemes, this scheme has higher packet forwarding rate, longer network life, and lower overhead. This further proves that the scheme is energy-saving, safe, dynamic, stable, and sustainable.

Keywords
Wireless sensor networks, cluster head, load balancing, authentication, security

Date received: 23 July 2019; accepted: 5 November 2020
Handling Editor: César Vargas-Rosales

Introduction
Wireless sensor network (WSN) consists of a variety of sensors distributed in space, which are used to monitor and sense the environment, such as temperature, humidity, light, sound, vibration, pressure, location, and even military terrain, and to transmit data to the base station (BS) through the network for processing.¹ For many researchers, WSN is a challenging research field, and it plays an important role in the Internet of Things, cloud computing, edge computing, and other fields.² WSNs may be deployed in harsh remote environments.³ The network is more vulnerable to some malicious attacks due to its unattended policy of deployment. Moreover, sensor nodes have limited resources and battery powered, so energy efficiency is one of the most critical problems faced by any WSN.

¹Institute of Information, Hunan University of Humanities, Science and Technology, Loudi, China
²Nanjing University of Posts and Telecommunications, Nanjing, China
³Corresponding author: Jiliang Zhou, Institute of Information, Hunan University of Humanities, Science and Technology, Dixing Road, Loudi City 417000, China. Emails: hnzhoujl@126.com; jlzhou@czie.edu.cn

Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
The energy consumption over a wireless network can be minimized by partitioning the sensor nodes as cluster heads (CHs). These CHs require more energy for data gathering and transmitting to BS. There are many routing protocols in WSN to distribute the energy consumption to CHs. CH is also responsible for authenticating the nodes that are allowed to join in a cluster and communicate with a network. Consuming more resources may result in a negative impact on lifespan of a network. Therefore, WSN need to use the smallest power consumption to achieve maximum security in order to achieve secure communication among nodes. When a network is said to have secure data transmission, it obviously results in energy consumption by providing authenticity, confidentiality, and integrity to the nodes. Hence, optimal energy utilization, thereby maximizing the network lifetime and insuring safety, is of prime importance in environments requiring high security.

However, when many sensors send a lot of data to the same CH, it is easy for the CH’s energy to be exhausted if it does not offload the neighbor CHs. The availability of CH services in data fusion, routing switching, and other technologies has also been greatly improved. At the same time, the energy-saving problem of sensor networks has also attracted people’s attention to the load balancing problem faced by CH. In order to solve the problem of load and energy balance, various new research technologies have been proposed, but they cannot solve the problem of CH security deployment and verification at the same time. Since CH deployment is usually in remote unattended scenarios, deployment is an important step before any load and energy balancing. In addition, because the network structure of CH deployment is distributed, load balancing can also be applied to distributed scenarios, which can be divided into dynamic load balancing and static load balancing. A major disadvantage of static load balancing is that it does not take into account the state of the target CH when determining load balancing. Dynamic load balancing uses a more real-time approach by considering the current load and energy on a single CH and recommending the target CH accordingly. This enables tasks to be dynamically allocated from overloaded CH to lower load or idle CH. Compared with the static method, the dynamic method is difficult to implement, but it poses a higher challenge and value for the dynamic solution of load and energy balance. In view of the above advantages, this article considers dynamic load and energy balance as well as security verification issues.

The node energy limitation requires that microcontroller has less compute power. Various encryption and decryption algorithms—such as Data Encryption Standard (DES), Rivest–Shamir–Adleman (RSA), Advanced Encryption Standard (AES)—are highly secure, but the nodes may require higher amounts of energy and additional computational capabilities. RC4 is a stream encryption algorithm, which originated in 1987 and is now considered unsafe. Compared with RC4, the key length of RC5 is 128 bits, but RC5 still needs only basic addition, exclusive, or cyclic operations and can be implemented on many hardware. Hence, encryption algorithms RC5 are well suited for implementation using WSN nodes. In this work, RC5 algorithm is considered.

A suitable network deployment in addition to energy-aware protocols is very important in the cases mentioned above. There are many solutions to implement authentication in existing distributed systems, but they are not suitable for CH deployment and authentication in WSNs. Due to the hostility of CH deployment, authentication becomes the main problem in identifying target CH for load and energy allocation. Existing deployment technologies authenticate network nodes without obtaining target loading information and residual energy. This poses new challenges in building dynamic load and energy balance through appropriate deployment and authentication. In this article, we propose a secure deployment scheme to select CH for load and energy balancing. The following is a summary of the main contributions of this proposed methodology:

- The proposed approach is an adaptive CH deployment and authentication technique which once initiated enables the CHs to authenticate each other using the BS credentials. We propose an adaptive CH deployment and authentication technology in WSNs. Once deployment is started, CH can use BS credentials to authenticate each other.
- The scheme also considers the real-time load and energy on each destination CH to achieve dynamic load and energy balance. During the deployment process, the scheme especially considers the minimization of communication overhead.
- The approach combines the above dynamic load and energy balancing data with deployment and authentication, and applies them to clustered WSNs. The performance of the scheme is evaluated and simulated, and the effectiveness and stability of the scheme are verified.

The rest of this article is structured as follows: section “Related works” discusses the related work of this article. Section “Proposed solution” elaborates on a secure dynamic load and energy balancing solution. Section “Security evaluation” presents the evaluation of our model. Section “Performance simulation” simulates the performance and efficiency of the proposed solution.
through simulation experiments. Section “Conclusion” gives the conclusions and future work.

Related works

This section briefly introduces the background research of related work, and analyzes the problems faced by the proposed solutions. When sensor nodes unload their tasks to neighbor CHs, there are differences in load, energy, location, especially security on different CHs. Because some CHs in high activity areas may be overloaded, while others in low activity areas may be idle or have very small workload, this can easily lead to extreme imbalance in task allocation. At present, researchers have put forward many strategies to solve the load balancing problem and security deployment and verification.

In order to overcome the excessive load in the high active area of WSNs, Palani et al. proposed a load balancing technology combined with routing protocol, which can effectively utilize multiple mobile sensors and use load balancing technology to balance the load of sensor nodes. However, there are congestion problems in the sink nodes of multiple mobile sensors, which increases the energy consumption of sensor nodes. Masdari and Naghiloo proposed a receiver selection algorithm based on distributed fuzzy logic, which can effectively prevent the congestion of sink nodes and load balancing between receivers. In order to reduce the discarding problem caused by the overflow of data queue in CH, Gherbi et al. proposed a hierarchical energy balanced multipath (HEBM) routing protocol, which balances the energy dissipation between sensor nodes and prolongs the lifetime of the network. Gherbi et al. also proposed a distributed energy efficient adaptive clustering protocol (DEACP), which has the characteristics of load balancing and self-adaptation. It can reduce the energy consumption of the whole network, balance the energy consumption between sensors, and prolong the lifetime of the network.

Neamatollahi et al. proposed a distributed energy efficient protocol using two techniques: local re-clustering and multi-criteria cluster formation. The protocol prolongs the lifetime of WSNs by decreasing clustering overhead. In order to minimize data redundancy and maximize network lifetime, Hassan and Abdellah proposed an enhanced clustering hierarchy (ECH) method to achieve energy efficiency of WSNs by sleeping and waking overlapping nodes and adjacent nodes. In order to design an efficient cluster for load balancing, Edla et al. proposed a new fitness function based on the number of gateway loads and overloaded nodes. The cluster algorithm based on shuffled complex evolution of particle swarm optimization (SCE-PSO) improved the lifetime of WSNs. In order to minimize the communication distance between CH and BS using multi-hop technology, Al-Zubi et al. designed a new method to select the initial CH, and proposed an improved energy-aware routing protocol for fixed cluster. The protocol balances the workload of all nodes in the network and reduces the energy consumption in the network. Samal et al. implemented a stable selection protocol (SEP) and a load balancing protocol, and proposed a new method to maximize the lifetime of WSN by combining energy sensing and load balancing in heterogeneous WSN, which achieves a uniform load distribution of each node’s energy capability. CALB algorithm is a fully distributed algorithm that only needs to communicate with adjacent sensors. Khoualene et al. proposed a cluster-based load balancing clustering algorithm, which optimizes other resources and prolongs the lifetime of no network. In order to optimize the transmission of network control messages and reduce network overhead, TTag et al. proposed a collaborative load balancing algorithm (CoLBA), which uses prediction method to avoid packet queue overflow and optimize the transmission of control messages.

Similarly, CH’s security authentication and deployment are as important as its load balancing in WSN. In order to develop a security authentication protocol for WSNs in coal mine safety monitoring, Kumari et al. proposed an authentication protocol to overcome security problems by authenticating users. Broadcast authentication is a basic security service in WSNs. Benzaid et al. proposed a pair-less identity-based signature verification based on the cooperation between sensor nodes. In order to implement the authentication of real-time applications in WSN, Gope et al. proposed a lightweight anonymous authentication protocol for real-time applications based on WSN to resist denial-of-service (DoS) attacks. Razzaque and Rizvi introduced the use of access control and authentication protocols for secure data aggregation (SDAACA), SDAACA protocol by dividing data into small pieces to hide opponent’s data, can timely and effectively identify malicious nodes. Zawaideh and Salamah proposed an effective malicious node detection (WT-MND) scheme based on weighted trust, which can detect malicious nodes in cluster WSNs. Khan et al. proposed an LEACH++ protocol based on intrusion detection framework, which can resist black holes and selective forwarding attacks. Guber et al. used resiliency, trust value, and neighborhood number to select a CH, and proposed a method of CH selection based on bio-inspiration and trust in WSN. Hima and Rama et al. proposed a lightweight secure and energy-saving LEACH protocol (LS-LEACH), which integrates broadcast authentication algorithms to authenticate users. Kumar and Umamakeswari proposed a general specification-based intrusion detection
model (SS-LEACH). SS-LEACH uses signature-based broadcast authentication to distribute code and data, which improves security, but requires high computational complexity. Afianti and Wirawan Suryani\textsuperscript{25} used threshold function to limit the number of hash iterations, and proposed a dynamic cryptographic puzzle (DCP). DCP increases the probability of attack and the complexity of signature-based DoS attack. In 2017, Mohit et al. proposed an authentication protocol for wireless sensors in vehicle communication, which guarantees the security of mutual authentication for vehicle communication users. In order to solve the security vulnerability of Mohit et al.’s scheme, S Yu et al.\textsuperscript{26} proposed a secure mutual authentication and anonymity security authentication protocol by changing dynamic parameters.

From the above discussion and analysis, these security authentication and deployment schemes do not adequately consider the load balancing of CHs, while the load-balanced schemes neglect the security authentication and deployment of CHs. For CH nodes in WSN, energy and security are equally important. Load balancing is to balance load and energy to secure neighbor nodes, while minimizing overhead as much as possible. This will be a more challenging design. Therefore, a new security authentication and deployment scheme is proposed in this article.

**Proposed solution**

**Network architecture**

In this section, a well-known clustering mechanism for WSNs is adopted, and a large number of sensor nodes are clustered hierarchically. As shown in Figure 1, each cluster consists of multiple sensor nodes and a CH. Suppose that all CHs need to be deployed remotely, and each CH uses code division multiple access (CDMA) technology to transmit data to BS. Assuming that all sensor nodes are secure, they can perceive all kinds of data and transmit data to their CHs in the form of time division multiple access (TDMA). In order to balance energy and reduce transmission delay, each channel may choose its secure neighbors to help data aggregation and fusion.

As compared with previous literature review, authentication is not considered in CH load balancing. Therefore, the author designs a cluster-based WSN architecture as shown in Figure 1, and proposes a new deployment scheme, which can not only verify the performance of CHs but also collect the current load information and residual energy of CHs before assigning tasks.

The following sections elaborate on the implementation process of load balancing and authentication scheme based on clustering network, including CH security deployment and authentication process, and how to combine authentication with dynamic load balancing. The notation used in the presented solution description is listed in Table 1.

**Secure deployment and authentication**

As per CH deployment in WSN, all the CH information must be stored and processed in the BS. In such a clustering WSN, CHs are intermediate layers that collect and fuse data for BS. The scheme assumes that BS is completely trusted, while CH of WSN is considered to be partially trusted at the edge of the network. Accordingly, the authors consider the BS for the initiation of the authentication process. This process starts with the assignment of an initial ID ($P_i$) linked to the

![Figure 1. Cluster-based WSN architecture.](image_url)
key \((K_i)\) and the session key \((K_s)\) for each CH during the CH deployment \((BS \rightarrow CHs \{P_i//K_i//K_s\})\). It is very important to store secret information and updated keys provided by BS. CHs use trusted modules, such as trusted platform module (TPM). Once initialized, a single CH initiates the deployment and validation process to validate other CH in the neighborhood. This can effectively prevent malicious CHs from being added to load and energy balancing. The key exchange of CHs is determined and updated by BS and CHs.

Assuming that CH-\(i\) starts the secure deployment and authentication process. It associates its ID with the relevant key, and then encrypts it with the session key sent by BS \((E_K(P_i//K_i))\). Then, CH-\(i\) broadcasts the ciphertext to all neighbor CHs in WSN. Once receive the ciphertext, CH will decrypt it using the session key \((D_{K_i}(P_i//K_i))\) issued by BS. A common session key \((K_i)\) is, however, initiated by the BS and provided to individual CHs and is also trusted by other CHs. When the target CH (CH-\(j\)) receives the source ID associated with the key, it checks them with the BS to verify the authenticity of the source CH \((CH - j \rightarrow BS \{E_K(P_i//E_{K_i}(P_i))\})\). If BS decrypts the ciphertext and found the ID of the CH-\(i\) by checking its own database, BS then encrypts the ACK packet \((E_K(P_i//K_i))\) with recipient’s secret key and sends to CH-\(j\). If CH-\(i\) is a node that has not passed the authentication, BS will also respond to the authentication request of CH-\(j\) and inform the verification result that it failed. Then CH-\(j\) connects the relevant key with its own ID and encrypts them using source relevant key such as \((E_K(P_i//K_i))\). Once receiving the ciphertexts, CH-\(i\) decrypts it with own secret key and get the \(P_i\). The ciphertext is of the format \(E_K(P_i//E_{K_i}(P_i))\), where \(P_i\) is encrypted with relevant key \(K_i\) of CH-\(i\). This integrates own ID to generate a ciphertext using session key of BS. CH-\(j\) decrypts the ciphertext using own session key after receiving acknowledgment from BS and then retrieves the relevant key of \(P_i\) and authenticates CH-\(j\). Once the process is authenticated, BS links \(P_i\) with the relevant key \((K_i)\) and encrypts with relevant key \((K_j)\) of CH-\(i\) to send it back to CH-\(i\). After receiving the ciphertext, CH-\(i\) decrypts it to get the key \((K_j)\), then compares with the relevant key from CH-\(j\). If the match is successful, that is, \(K_j = K_j\), CH-\(i\) binds the ID of CH-\(i\) and CH-\(j\) and encrypts them using the destination’s relevant key \((K_i)\). This combination of cipher packets is sent to CH-\(j\), which confirms that both CH-\(i\) and CH-\(j\) have now been verified by load and energy balancing. The proposed authentication model adopts a method of obtaining authentication information directly from secure BS. Algorithm 1 gives the mutual authentication steps between CHs. Followed by, CH-\(i\) links relevant key between it and CH-\(j\) \((K_j)\) with its ID \(P_i\) and encrypts them with CH-\(j\) key such as \((E_{K_j}(P_i//K_j))\), and CH-\(j\) responds to CH-\(i\) with ACK \((E_{K_j}(P_i//P_j))\). Thus, the newest real-time session key between CH-\(i\) and CH-\(j\) is generated and stored separately. Finally, there are different session keys between any two neighbor nodes. The process information of security authentication is shown in Figure 2.

### Secure load balancing

On the basis of security authentication and deployment, we continue to develop a lightweight secure load and energy balancing (SLEB) scheme. The proposed scheme uses four parameters \(m_l\) and \(m_c\), \(e_l\) and \(e_c\) to check the load and energy for each CH, where the current CH’s load of is defined by \(m_l\) and the computing capacity is defined by \(m_c\); where the current CH’s residual energy is defined by \(e_l\) and the CH’s total energy is defined by \(e_c\). To calculate the current load on a CH, the parameters \(q_m\) and \(q_e\) are used, where equations (1) and (2) are satisfied, respectively

\[
q_m = \frac{m_l}{m_c} \leq 0.7
\]

\[
q_e = \frac{e_l}{e_c} \geq 0.5
\]

If a CH CH-\(i\) overloads, it will broadcast a control package containing its load information \(L_i\) and its own ID \((P_i)\) and residual energy \((R_i)\), as a request to other adjacent CHs. Each CH receives a load and energy balancing request package from its neighbor CHs. Here, the \(P_i\) defines the ID of the CH-\(i\) sending the request and \(L_i\) defines the received load information. A adjacent CH (called CH-\(j\)) checks it by comparing the received ID with its own database. If a match is found, CH-\(j\) will look for load information and residual energy from the control packets, however. If a match is found, CH-\(j\) will look for load and energy information to avoid a possible network attack.

While recipient CH-\(j\) processes the load balancing information at CH-\(i\), it checks the load and residual energy information using value of parameters \(q_m\) and
If the value of parameter $q_m$ is less than or equal to 0.7 (i.e. close to 30% of free processing resources) and the value of parameter $q_e$ is more than or equal to 0.5 (i.e. close to 50% of free energy resources), and the energy and computing resource available index (i.e. $m_2$ and $e_2$) to perform this application task from CH-$i$, then CH-$i$ initiates to prepare the positive ACK to CH-$i$. If residual energy and resource of recipient CH-$j$ is more than the required resource to process the application task, then CH-$j$ sends the reply packet to CH-$i$. Otherwise, CH-$j$ never responds to CH-$i$.

If all conditions satisfy, CH-$j$ sends response packet including own identity ($P_j$), relevant key ($K_j$) and resource parameters $q_m$ and $q_e$. Finally, the response packet is encrypted with the session key $K_{ij}$ of CH-$i$ ($E_{K_{ij}}(P_j // K_j // q_m // q_e)$) and sends it to the CH-$i$ for next processing of load balancing. Once receiving the ciphertexts, CH-$i$ applies corresponding session key that is, $K_{ij}$ to decrypt the data packets ($D_{K_{ij}}(P_j // K_j // q_m // q_e)$). CH-$i$ then authenticates the source ID ($P_j$) with its own database to find matching ID. If the match is successful, CH-$i$ is then compared with the previously saved key ($K_j$) of the packets decrypted. If the match is also successful, CH-$i$ accepts the response information ACK from CH-$j$, otherwise it is ignored to avoid network attack. In the same way, CH-$i$ receives a large number of replies from different CH in the region. CH-$i$ compares the values of

---

**Algorithm 1.** Secure deployment and authentication.

**Input:** BS initialized the shared key and identification to all CHs

**Output:** all the CHs are authenticated with each other to securely balance the load and energy consumption, and generate shared common keys between any two neighbor nodes.

**Procedure:**

1. BS unicasts the individual CH's ID, key and common shared key:
   
   $$BS \rightarrow CH-i(P_i // K_i // K_s)$$

2. CH-$i$ broadcasts the generated packets: $E_{K_i}(P_i // K_i)$

3. Recipient CH-$j$ verifies the authenticity of CH-$i$: $CH-j \rightarrow BS \{E_{K_i}(P_i // E_{K_i}(P_i))\}$

4. Check CH-$i$ in BS’ database

5. If BS found CH-$i$ is a authenticated node, then BS responds the ACK to CH-$j$: $E_{K_i}(P_i // K_i)$

6. Else the BS responds to CH-$j$ declining the authentication request.

7. CH-$j$ decrypts the packet: $D_{K_j}(P_i // K_i)$ it to find the key ($K_j$)

8. CH-$j$ matches $K_j$ and $K_j$. If $K_j = K_j$, then send the CH-$i$: $E_{K_i}(P_i // K_i)$

Else don’t respond.

9. CH-$i$ authenticates CH-$j$ with BS by following step 3, then GO TO STEP 3.

10. CH-$i$ generates new shared key with CH-$j$ ($K_{ij}$), send to CH-$j$:

    $$E_{K_i}(P_i // K_i)$$

11. CH-$j$ responds to CH-$i$ with ACK: $E_{K_{ij}}(P_i // P_i)$. 

---

**Figure 2.** Stepwise for the secure deployment and authentication of CHs.
Algorithm 2. Load and energy balancing technique.

Input: CH-i initialized the shared key and identification with all neighbor CHs
Output: CH-i found the neighbor CH with the most energy and the least load as load balancing

Procedure:
1. If (CH-i is overloaded) then broadcast packet: \((P_i,L_i)\).
2. CH-j (neighbor of CH-i) checks the received ID: \(P\)
3. If \((P_i\) is in database), then CH-j executes step 4.
Else ignore the packet: \(\{P_i\}\).
4. If \((q_m < 0.7 \& q_e > 0.5 \& L_i \ll m_e - m)\) then response packet: \((\{E_{K_i} (P_i/K_i/q_m/q_e)\})\).
Else ignore the packet: \(\{P_i\}\).
5. CH-i performs \(D_{K_i} (P_i/K_i/q_m/q_e)\)
6. If \((P_i/K_i)\) is in database then CH-i executes step 7.
Else CH-j doesn’t continue to process the packet.
7. CH-i selects CHs with maximum value of \(q_m\) in all neighbors.
8. If (exist more than two maximum value of \(q_m\)) then selects the CH with minimum value of \(q_m\) for load balancing.

Theorem 1. The malicious attacker Ma cannot read the secret vouch of the CH to impersonate the CH to participate in the load and energy balance.

Proof. According to the above definition of the authenticity of the TPM module (the security module of the CH), the attacker Ma cannot obtain secret information such as \(P_i, K_i, K_s\) initiated by the BS. All security information for authentication is initiated by the BS during CH deployment. When the CH starts mutual authentication, they use the BS session key \(K_s\) to encrypt the initial authentication packet \(E_K_i (CH_i/K_i)\), and then use CHs \(K_{ij}\) each associated key is used to encrypt the communication packet. During authentication, the keying technique used follows RC5 encryption, which is a lightweight, symmetric block cipher with a variable-length key that takes years to decipher.

Thus, after the TPM attribute, it is almost impossible to get a process or key from the security module. Therefore, attacker Ma is unable to impersonate CH to participate in load and energy balance.

Security evaluation

By combining theoretical analysis with formal verification, the proposed security deployment and authentication mechanism are evaluated, and the details are discussed as follows.

Security proof

Definition (authentication attacker). Authentication attacker “Ma” can initiate attacks on the authenticity of identity, can impersonate the authenticated CH, and can also start the load and energy balancing process.20

Definition (confidentiality attacker). Confidentiality attacker “Mc” refers to the unauthorized person who can identify information when load balancing between CH-i and CH-j.20

Definition (attacker of integrity). Information integrity attacker “Mi” can monitor load information and residual energy, and try to access or modify messages between CHs.20
Theorem 3. SLEB is a dynamic, sustainable and efficient safe operation mechanism by choosing CH with less load and more energy to balance the load.

Proof. SLEB uses a BFS (breadth first search) solution to balance the load between CHs in the sensor network to ensure efficient operation. Wherein, the searched network is defined as G (V, E), and N(v) defines the number of neighboring nodes, where V is CH. R(CH₁, CH₂, ..., CHₙ) is the set of neighboring CHs from the network. CH ∈ V, all CHs are within the range of load and energy sharing. The source CH broadcasts overload information so that the receiver CH may share the load and energy. Next, the CH receives x responses, where 1 ≤ x ≤ n. As follows the technique proposed by the authentication solution during the initial broadcast (refer to Algorithm 1), CH-i receives the response with \( E_{K_j}(P_j/K_j/q_m/q_e) \), reflecting the security of SLEB. The data packet queries real-time information about the current load \( (q_m) \) of the recipient CH and the current energy \( (q_e) \) after identifying the authenticity of the received data packet. It reflects the combination of SLEB’s security and dynamic data, making SLEB run more efficiently. More importantly, the CH will respond to this request only when the receiver has sufficient resources \( (q_m ≤ 0.7 \text{ and } q_e ≥ 0.5) \) to handle multiple loads, ensuring the subsequent operational capability of the CH. This in turn ensures the stability and sustainability of SLEB.

Moreover, the symmetric encryption method adopted in this mechanism has the advantages of less computation, faster encryption, and decryption speed, and its efficiency is more than 1000 times of asymmetric encryption, which greatly improves the processing speed of packets and the response time of nodes, consequently saves the energy of nodes, and prolongs the lifetime of nodes. In summary, this mechanism not only ensures the security and efficiency of load balancing but also ensures that CH with less load and more residual energy can actively share its load and energy.

Forward secrecy

By following the standard symmetric key encryption algorithm, the initial identity of CH is verified. Since authentication occurs only once at the beginning of CH’s initialization, CH identical key is used to verify the CH’s authentication of the receiver. However, if an intruder acquires the key for authentication, it will not be useful after the initial authentication. After the deployment of CH is completed, fresh and real-time symmetric key pairs will be generated among different neighbors, so that the information of CH in load and energy balance can be securely encrypted.

Formal security verification

The Scyther simulation environment can verify the security of the protocol. The authors write simple code with the security protocol description language (SPDL) of Scyther to test the proposed security scheme (SLEB). Two roles are defined: CH-i and CH-j, and CH-i initiates authentication and sends data to CH-j. The authentication process starts with sending a packet from CH-i to CH-j. Next, CH-j responds to the load information and residual energy sent by CH-i. Because of the existence of malicious attackers for authentication in this authentication scenario, a malicious attack packet is sent to CH-j in an attempt to start the load and energy balancing process. The scenario runs 200 times and checks every 10 times to verify any possible attack on identity authenticity.

For this particular test scenario, the authors ignore all kinds of non-authentication attacks that may be included in the attack model and focus only on the authentication attacks of CH nodes. It is assumed that the attacker can observe and replicate the communication between CH and try to achieve load and energy balance between CH. The solution proposed by the authors is to use the trusted modules mentioned above (such as the TPM of CH) to store sensitive information, including updating the key process and key.

The authors run 200 iteration experiments in the Scyther environment, and check them every 10 times. Throughout the runtime, tests found that SLEB did not cause any successful authentication attacks. Figure 3 shows SLEB verification result page in Scyther environment, demonstrating that the proposed security solution SLEB has the ability to withstand authentication attacks.

Performance simulation

The purpose of the experiments was to evaluate the performance of the SLEB compared to the other protocols or technology such as S-LEACH, MS-LEACH, and SS-LEACH. To simplify the simulation experiment, we generated random nodes and defined some of them as malicious nodes. In the following sections, we measured the packet forwarding ratio, protocol overhead, and network lifetime by malicious attacks during data forwarding for CHs in WSNs, and for different scenarios, we then show the following simulation results.

Simulation metrics

In order to evaluate the performance of security load and energy balance mechanism in the presence of malicious nodes that affect network performance, we simulated SLEB, S-LEACH, MS-LEACH, and SS-LEACH on the network simulator NS-3.28. The authors
consider the generation of a variety of different numbers of sensors; each sensor node has a constant transmission range of 20 m, and all sensor nodes are stable and no node movement. Random configuration from 50 to 500 sensor nodes randomly placed in the target area of 200×200 m², the lower left corner as (0,0), BS positioning at a fixed point (200, 200) in the center of WSN. Each round sends 20 packets to CH, and each time we change the scenario file to test the performance of S-LEACH, MS-LEACH, and SS-LEACH when different numbers of malicious nodes appear. The energy model and other simulation parameters are summarized in Tables 2 and 3.

### Table 2. Simulation parameters.

| Parameter                  | Value                                      |
|----------------------------|--------------------------------------------|
| Simulation time            | 600 s                                      |
| Number of nodes            | 50, 100, 150, 200, 250, 300, 350, 400, 500 |
| Simulation area            | 200×200 m²                                  |
| Number of runs per scenario| 10 times                                   |
| Base station positions     | Corner (5, 5) and center (100, 100)        |
| Number of CHs              | 3, 6, 9, 12, 15, 18, 21, 24, 27, 30        |
| Round time                 | 50 s                                       |
| Data signal size           | 512 bytes                                  |
| Channel BW                 | 1 Mbps                                     |

CH: cluster head; BW: bandwidth.

### Table 3. Energy model parameters.

| Parameter                  | Value |
|----------------------------|-------|
| Initial energy             | 10 J  |
| Electronics energy         | 50 nJ/bit |
| Receive threshold          | 1 nJ  |
| Success threshold          | 6 nJ  |
| Data aggregation energy    | 5 nJ/bit/signal |

Packet forwarding ratios

In this scenario, for every test case, we increased some malicious CHs into the WSN. In Figure 4, there are some malicious CHs from 3 to 30 presented in the WSN, and we show the packet forwarding rates of MS-LEACH, S-LEACH, and SS-LEACH protocols. From the figure, all the time the SLEB achieves better packet forwarding ratio than the other schemes MS-LEACH, S-LEACH, and SS-LEACH protocols. This is because in SLEB, malicious CHs cannot join into the network through authentication, the identity of all malicious nodes cannot be verified, and sending data packets will be ignored by normal CH. Therefore, the packet forwarding rate of SLEB is higher than that of other protocols.

With the increase in the number of malicious nodes, the packet forwarding rate of the four protocols decreases, because malicious nodes may carry out
various destructive activities. Instability of the packet forwarding ratio for S-LEACH, MS-LEACH, and SS-LEACH protocols is due to the instability of the environment including malicious nodes. Especially, the packet forwarding ratio for S-LEACH protocol descends obviously. That malicious CHs may occupy a key position in data transmission because some malicious CHs are randomly selected. In SLEB, malicious CHs cannot pass authentication and decrypt data packets due to security authentication and deployment mechanisms. In addition, SLEB will spend a certain amount of time establishing connections and possibly processing packets sent by malicious nodes, and the packet forwarding rate of SLEB will decrease slightly.

**Protocol overhead**

The overhead of SLEB is a key parameter to prove whether SLEB is lightweight. We have performed the experiments to measure the overhead of SLEB. Figure 5 plots the average number of packets exchanged during neighbor discovery phase as a function of the number of malicious CHs in WSNs. For a given number of nodes, 10 different network topologies are generated at random, and WSNs of each topology placed 6% of malicious nodes. The average number shown by asterisks in Figure 5 is the average number of packets exchanged over these 10 different network topologies.

We compare SLEB with S-LEACH, MS-LEACH, and SS-LEACH protocols in Figure 5. It is clear that SLEB sends more packets than S-LEACH, MS-LEACH, and SS-LEACH protocols, and the difference increases with increasing numbers of nodes in the network. The overhead for the SLEB goes up smoothly and increases from 13% to 20% while LEACH and S-LEACH protocols increase from 14% to 28%. In SLEB, when deployment is complete, most broadcasting of control packages is done only in load balancing. When malicious nodes are detected, CH does not need any additional overhead and ignores the message directly. In LEACH and S-LEACH protocols, when a misconduct is found, a third party is needed to deal with it, which consumes communication overhead. To do this, the CH simply sends the same interest with a monotonically increasing timestamp attribute. This is necessary because interests are not reliably transmitted throughout the network. The refresh rate is a protocol design parameter that trades off overhead for increased robustness to lost interests. Further flooding is needed to find new path. This difference is attributed to the overhead involved in dealing with security and intrusion detection issues.

**Network lifetime**

One of the most important performance parameters in WSN is the network lifetime. In Figure 6(a), compared with S-LEACH, the network lifetime of SLEB increases by more than 25%, and is much longer than that of SS-LEACH and MS-LEACH. Although the rules for enforcing specific paths are different, S-LEACH, SS-LEACH, and MS-LEACH always use the same path for all communications between the same source and BS. The direct consequence is that the nodes on this particular path may quickly run out of energy, and SLEB will choose the neighbors who have the ability to share the load and balance according to their own and neighbor’s energy situation, and have a dynamic array of neighbor energy, which can avoid the rapid energy exhaustion of a single node.

Figure 6(b) shows a simulation of the network lifetime in WSN when there are 6% malicious nodes in the...
network. As can be seen from the graph, the network lifetime of S-LEACH, SS-LEACH, and MS-LEACH decreased significantly, while that of SLEB decreased slightly. The lifetime of SLEB was 37% longer than that of S-LEACH, SS-LEACH, and MS-LEACH. The authors analyze that when CH detects malicious, S-LEACH, SS-LEACH, and MS-LEACH protocols must choose a new path, and the communication load propagates between a small number of available paths. However, SLEB can reject malicious CH and resist its attack. Furthermore, according to Algorithm 2, the result of load balancing is the routing transfer of multi-path shared packets.

Conclusion
Because of the limited resources of WSNs, network security and network life cycle are very important. In this article, a secure load balancing scheme for clustered WSNs is proposed, which combines load balancing and security verification effectively, and achieves CH authentication while balancing CHs load. SLEB also considers the real-time load and energy on each destination CHs to achieve dynamic balance. The SLEB we proposed is completely different from the well-known LEACH and S-LEACH protocols and other routing protocols that contain security. SLEB not only implements the identity authentication of CH and the confidentiality and integrity of all messages but also implements the identity authentication and packet encryption between any two CHs in data load balancing. SLED reduces the load of CHs while authenticating CHs. It not only effectively maintains the energy balance of the whole network and prolongs the network lifetime but also successfully improves the security mechanism, guarantees the real-time security, and low overhead. The proposed load balancing solution has been theoretically analyzed and experimentally evaluated and simulated from multiple perspectives. Based on the results of performance evaluation and comparison, we can conclude that the proposed solution is safe, dynamic, stable, and sustainable. The simulation results show that the performance evaluation of SLEB is better than that of S-LEACH, SS-LEACH, and MS-LEACH in the presence of malicious nodes. Our protocol surpasses S-LEACH, SS-LEACH, and MS-LEACH in terms of packet forwarding rate, protocol overhead, network lifetime, and nodes attacked during data forwarding.

However, we still have a lot of research works to continue to explore. For the data fusion mechanism of WSNs, we can refine the characteristics of data and achieve load balancing more efficiently. For the security mechanism of WSNs, we can refine the specific attack behavior of malicious nodes, so as to better. Load balancing combines with security. In addition, according to the characteristics of wireless sensor, the integrated test platform of data and security is also worth studying.

Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Natural Science Foundation of China under grant no. 61562073, the
Ministry of Education Information Project (no. 2018LXB0222), and our school projects which are “Cloud Product Solution for Supply Chain Business Collaboration” (no. 11130300117028) and “Research on Some Crucial Issues of Energy Optimization in Wireless Sensor Networks” (no. 111309001170014).

**ORCID ID**

Jiliang Zhou https://orcid.org/0000-0001-8797-2715

**References**

1. Rawat P, Singh KD, Chaouchi H, et al. Wireless sensor networks: a survey on recent developments and potential synergies. *J Supercomput* 2014; 68(1): 1–48.

2. Jaladi AR, Khithani K, Pawar P, et al. Environmental monitoring using wireless sensor networks (WSN) based on IoT. *Int Res J Eng Technol* 2017; 4(1): 1371–1378.

3. Yassen MB, Aljawaerneh S and Abdulraziq R. Secure low energy adaptive clustering hierarchal based on internet of things for wireless sensor network (WSN): survey. In: *Proceedings of the IEEE international conference on engineering & MIS, Agadir, Morocco*, 22-24 September 2016, pp. 1–9. New York: IEEE.

4. El-Saadawy M and Shaaban E. Enhancing S-LEACH security for wireless sensor networks. In: *Proceedings of the IEEE international conference on electro/ information technology (EIT)*, Indianapolis, IN, 6–8 May 2012, pp. 1–6. New York: IEEE.

5. Palani U, Alamemangai V and Nachiappan A. Hybrid routing and load balancing protocol for wireless sensor network. *Wirel Netw* 2016; 22(8): 2659–2666.

6. Masdari M and Naghiloof F. Fuzzy logic-based sink selection and load balancing in multi-sink wireless sensor networks. *Wirel Pers Commun* 2017; 97(2): 2713–2739.

7. Gherbi C, Aliouat Z and Benmohammed M. An adaptive clustering approach to dynamic load balancing and energy efficiency in wireless sensor networks. *Energy* 2016; 114: 647–662.

8. Gherbi C, Aliouat Z and Benmohammed M. A novel load balancing scheduling algorithm for wireless sensor networks. *J Netw Syst Manag* 2019; 27(2): 430–462.

9. Neamatollahi P, Taheri H, Naghibzadeh M, et al. A distributed clustering scheme for wireless sensor networks. In: *Proceedings of the 2014 6th conference on information & knowledge technology*, Shahrood, Iran, 27-29 May 2014. New York: IEEE.

10. Hassan EA and Abdellah N. Fuzzy logic based clustering algorithm for wireless sensor networks. *Int J Fuzzy Syst Appl (IJFSA)* 2017; 6(4): 63–82.

11. Edla DR, Kongara MC and Cheruku R. SCE-PSO based clustering approach for load balancing of gateways in wireless sensor networks. *Wirel Netw* 2019; 25(3): 1067–1081.

12. Al-Zubi R, Abdelsalam N, Atieh A, et al. LBCH: load balancing cluster head protocol for wireless sensor networks. *Informatica* 2018; 29(4): 633–650.

13. Samal SR, Bandopadhyaya S, Pathy A, et al. An energy efficient head node selection for load balancing in a heterogeneous wireless sensor network. In: *Proceedings of the 2018 52nd Asilomar conference on signals, systems, and computers*, Pacific Grove, CA, 28-31 October 2018, pp. 1428-1433. New York: IEEE.

14. Khoulalene N, Bouallouche-Medjkoune L, Aissani D, et al. Clustering with load balancing-based routing protocol for wireless sensor networks. *Wirel Pers Commun* 2018; 103(3): 2155–2175.

15. TTall H, Chalhoub G, Hakem N, et al. Load balancing routing with queue overflow prediction for WSNs. *Wirel Netw* 2019; 25(1): 229–239.

16. Kumari S and Om H. Authentication protocol for wireless sensor networks applications like safety monitoring in coal mines. *Comput Netw* 2016; 104: 137–154.

17. Benzaid C, Lounis K, Al-Nemrat A, et al. Fast authentication in wireless sensor networks. *Future Gener Comput Syst* 2016; 55: 362–375.

18. Gope P, Lee J and Quek TQS. Resilience of DoS attacks in designing anonymous user authentication protocol for wireless sensor networks. *IEEE Sens J* 2017; 17(2): 498–503.

19. Razaque A and Rizvi SS. Secure data aggregation using access control and authentication for wireless sensor networks. *Comput Secur* 2017; 70: 532–545.

20. Zawaideh F and Salamah M. An efficient weighted trust-based malicious node detection scheme for wireless sensor networks. *Int J Commun Syst* 2019; 32(3): 3878–3891.

21. Khan FA, Farooqi AH and Derhab A. A comprehensive security analysis of LEACH++ clustering protocol for wireless sensor networks. *J Super Comput* 2019; 75(4): 2221–2242.

22. Gaber T, Abdelwahab S, Elhoseny M, et al. Trust-based secure clustering in WSN-based intelligent transportation systems. *Comput Netw* 2018; 146: 151–158.

23. Hima Bindu A and Rama Prasad VV. Secure energy efficient LEACH (EE_LEACH) protocol for wireless sensor networks. *I-manager’s J Wirel Commun Netw* 2016; 5(2): 12–22.

24. Kumar SR and Umamakeswari A. SSLEACH: specification based secure LEACH protocol for Wireless Sensor Networks. In: *Proceedings of the 2016 International conference on wireless communication, signal processing and network*, Chennai, India, 23-25 March 2016, pp.1672-1676. New York: IEEE.

25. Afianti F and Wirawan Suryani T. Dynamic cipher puzzle for efficient broadcast authentication in wireless sensor networks. *Sensors* 2018; 18(11): 4021–4043.

26. Yu S, Lee J, Lee K, et al. Secure authentication protocol for wireless sensor networks in vehicular communications. *Sensors* 2018; 18(10): 3191–3211.