Hindawi
Security and Communication Networks
Volume 2021, Article ID 9082570, 12 pages
https://doi.org/10.1155/2021/9082570

Research Article

Detection and Location of Malicious Nodes Based on Homomorphic Fingerprinting in Wireless Sensor Networks

Zhiming Zhang, Yu Yang, Wei Yang, Fuying Wu, Ping Li, and Xiaoyong Xiong

School of Software, Jiangxi Normal University, Nanchang 330027, China

Correspondence should be addressed to Zhiming Zhang; zzm_9650@163.com

Received 8 July 2021; Revised 23 August 2021; Accepted 30 August 2021; Published 24 September 2021

Academic Editor: Ahmed A. Abd El-Latif

The current detection schemes of malicious nodes mainly focus on how to detect and locate malicious nodes in a single path; however, for the reliability of data transmission, many sensor data are transmitted by multipath in wireless sensor networks. In order to detect and locate malicious nodes in multiple paths, in this paper, we present a homomorphic fingerprinting-based detection and location of malicious nodes (HFDLMN) scheme in wireless sensor networks. In the HFDLMN scheme, using homomorphic fingerprint and coding technology, the original data is divided into n packets and sent to the base station along n paths, respectively; the base station determines whether there are malicious nodes in each path by verifying the validity of the packets; if there are malicious nodes in one or more paths, the location algorithm of the malicious node is implemented to locate the specific malicious nodes in the path; if all the packets are valid, the original data is recovered. The HFDLMN scheme does not need any complex evaluation model to evaluate and calculate the trust value of the node, nor any monitoring nodes. Theoretical analysis results show that the HFDLMN scheme is secure and effective. The simulation results demonstrate promising outcomes with respect to key parameters such as the detection probability of the malicious path and the locating probability of the malicious node.

1. Introduction

With the rapid development of the Internet of Things, wireless sensor networks (WSNs) are not only widely used in transportation, agriculture, home furnishing, military, environmental monitoring, and other fields [1] but also used in smart city environments [2], smart grid [3], and smart healthcare system [4], and Underwater Sensor Networks (USNs) have become widespread and are being deployed in a wide range of applications ranging from harbor security to monitoring underwater pipelines and fish farms [5] recently. Since WSNs are constructed by a large number of sensor nodes in a wireless and multihop way, and the sensor nodes are restricted by calculation, storage, and communication, they are easy to be captured as malicious nodes by attackers. The existence of malicious nodes is a great threat to the network; by manipulating these malicious nodes, attackers can launch a variety of internal and external attacks [6], for example, monitoring the important confidential information passing through these malicious nodes, injecting a large number of false data into sensor networks, destroying the normal data aggregation process by tampering with the data, launching various DoS attacks, and so on [7, 8]. Malicious nodes in multipath are more harmful because malicious nodes will send false data or pollution data to nodes in multiple paths at the same time, which is easy to cause the pollution data to continue to spread, thus consuming a large number of valuable resources of intermediate forwarding nodes and ultimately shorten the life cycle of the entire wireless sensor network; therefore, it is very important to detect, locate, and isolate the malicious nodes in multipath.

Detection of malicious nodes has always been a hot topic in wireless sensor networks; many scholars have proposed some effective detection schemes of malicious nodes. The current detection schemes of malicious nodes mainly focus on how to detect and locate malicious nodes in a single path;

however, for the reliability of data transmission, many sensor data are transmitted by multipath in wireless sensor networks [9–13]. In order to detect and locate malicious nodes in multiple paths, in this paper, we present a homomorphic fingerprinting-Based detection and location of malicious nodes (HFDLMN) scheme in wireless sensor networks. In the HFDLMN scheme, if the source node wants to send sensor data to the base station (BS), it divides the sensor data into n fragments and then encodes the n fragments to n new fragments; then, using homomorphic fingerprint technology, n packets are generated and sent to the base station along n paths, respectively. After receiving n packets with the same data number from n paths, the base station determines whether there are malicious nodes in each path by verifying the validity of the packets; if there are malicious nodes in one or multiple paths, the location algorithm of the malicious node is implemented to locate the specific malicious nodes in the path; if all the packets are valid, the original data will be recovered.

The main contributions of this paper are as follows. (1) A homomorphic fingerprinting-based detection and location of malicious nodes (HFDLMN) scheme in wireless sensor networks is presented; using homomorphic fingerprint and coding technology, the HFDLMN scheme can detect and locate malicious nodes in multiple paths. (2) In order to detect and locate malicious nodes, the HFDLMN scheme does not need any complex evaluation model to evaluate and calculate the trust value of the node, nor any monitoring nodes. (3) The HFDLMN scheme can resist the malicious node interfere with the base station to detect malicious nodes. (4) Theoretical analysis results show that the HFDLMN scheme is secure and effective, and the simulation results show it can detect and locate malicious nodes with high probability by sending a small number of packets.

The rest of the paper is organized as follows. Section 2 introduces the related work. Preliminaries and the system model are described in Section 3. The HFDLMN scheme is described in Section 4. Proof and analysis of related theorems are described in Section 5. Security analysis is described in Section 6. The performance evaluation is implemented in Section 7. Section 8 concludes this paper.

2. Related Work

At present, scholars have done a lot of work for detecting the malicious nodes in wireless sensor networks and have proposed some effective detection schemes. These schemes can be divided into multihop acknowledgment-based detection schemes, trust evaluation-based detection schemes, and statistics classification-based detection schemes.

Balakrishnan et al. [14] proposed a two-hop acknowledgment detection scheme (TWOACK) based on the checkpoint node. In the TWOACK scheme, each node in the forwarding path is the checkpoint node. If a node i receives a packet, it will send an acknowledgment packet to node j that two hops away from it. If node j does not receive the acknowledgment packet, it suspects the link between i and j to be a malicious link and sends a warning to the source node. However, the TWOACK scheme greatly increases conflict and collision of network messages. To solve this problem, Xiao et al. [15] proposed a multihop acknowledgment-based detection scheme (CHEMAS). In the CHEMAS scheme, some nodes in the path from the source node to the base station are randomly selected as checkpoint nodes. After the checkpoint node receives a packet, it will send an acknowledgment packet to its upstream node. If an intermediate forwarding node in the path does not receive the specified number of acknowledgment packets, it will suspect that its next-hop node is a malicious node and send a warning to the source node. Although CHEMAS can greatly reduce the conflict and collision of network messages, if two or more malicious nodes are selected as checkpoint nodes in the CHEMAS scheme, the collusion of these malicious nodes will make the CHEMAS scheme invalid. In order to solve this problem, Liu et al. [16] proposed a new scheme based on multihop acknowledgment mechanism (PHACK). In the PHACK scheme, in order to detect and locate malicious nodes, each node in the forwarding path not only needs to forward normal packets but also needs to generate an acknowledgment packet for each packet and send it to the source node along a different path. However, these schemes based on multihop acknowledgment need to transmit a large number of confirmation packets, which will increase high communication overhead and greatly reduce the network life.

To improve the effect of malicious nodes detection, Yang et al. [17] proposed a malicious node detection model based on reputation with enhanced low energy adaptive clustering hierarchy, MNDREL. Based on the enhanced routing protocol, the cluster head nodes are selected and other nodes form different clusters by choosing the corresponding cluster head. By analyzing the reputation value for the parent node evaluated by the child node, the malicious nodes in the network are effectively identified. The MNDREL model outperformed in detecting malicious nodes in WSN with lower false alarm rate; however, the real-time performance of the MNDREL model has to be improved. Xiao et al. [18] proposed a sensor network reputation model based on Gaussian distribution (GRFSN). In this model, the trust value of each node is obtained by calculating the weight sum of direct reputation and indirect reputation, and finally, compared with the trust threshold, if the trust value of the node is less than the trust threshold, the node is a malicious node. This scheme only needs to determine a trust threshold, but the trust threshold is static, and the misjudgment rate of normal nodes being judged as malicious nodes is high. In order to detect the untrusted nodes in the network quickly and effectively and ensure the reliable operation of the network, Zheng et al. [19] proposed a network security mechanism based on trust management to deal with the threats faced by WSNs (DNSMTM). Based on the trusted access of nodes, this mechanism firstly calculates the local trust degree of nodes according to existing interaction behavior and further obtains the comprehensive trust degree of nodes that can reflect the trust degree of nodes, and the detection of malicious nodes is carried out according to the comprehensive trust degree of nodes. The mechanism can effectively detect malicious nodes, with a higher detection

rate, and reduce the energy consumption of nodes. Liao et al. [20] proposed a hybrid strategy monitoring-forwarding game detection scheme to detect selective forwarding attack (MSGSFS). In this scheme, a set of strategies is constructed by integrating factors such as packet loss, data corruption, and forwarding delay. The data sending node and its one-hop neighbor nodes select strategies from the set to perform the monitoring-forwarding game and collect the routing trust value of the suspicious node. In order to locate and isolate malicious nodes in the cluster, a distributed watchdog is run on each cluster head node to monitor and record the forwarding behavior of its one-hop neighbor cluster head node. This scheme can effectively alleviate selective forwarding attack in wireless sensor networks and has less energy consumption. Zhou et al. [21] proposed an improved trust evaluation model based on Bayesian and Entropy (ITEMBB). In this model, the direct trust value of the node is first calculated, and if the direct trust value is not reliable enough, the indirect trust value of the node is calculated. By integrating the direct trust value and the indirect trust value, a comprehensive trust value is obtained, and entropy is used to assign a greater weight to highly trusted nodes. To a certain extent, the model overcomes the limitations of subjective weight allocation, but the problem of static reputation value has not been solved. Zhou et al. [22] combined the neighbor node monitoring and watchdog mechanism to propose a cluster-based selective forwarding attack detection scheme (SMCSF). In this scheme, the nodes in the cluster are divided into three types: cluster head nodes, monitoring nodes, and cluster member nodes; by selecting the monitoring node in the cluster, the monitoring node performs the calculation and adjustment of the comprehensive reputation of the cluster head nodes and cluster member nodes in the cluster. And in this scheme, the monitoring nodes are not only responsible for calculating and adjusting the reputation of the node and judging and detecting malicious nodes in the cluster but also responsible for monitoring whether the cluster head node has malicious behaviors such as data tampering or packet loss during the data forwarding process. Although this scheme can quickly and accurately locate malicious nodes, the responsibility of monitoring nodes is too heavy.

Silva et al. [23] proposed a detecting scheme of malicious nodes based on statistics (IDSBS). The scheme matches and detects the abnormal behavior of nodes through a series of predetermined rules. Because there is no interaction between nodes, the false detection rate of the system is high in the initial stage. Liu et al. [24] proposed a detecting scheme of malicious nodes based on classification (MCMND). In this scheme, first, the multiple attributes of the node are modeled, and then, the known sensor nodes are learned by the multiple classification method based on likelihood. The posterior probability is used to generate a classifier, for any unknown type of nodes, the nodes are classified according to the class with the maximum posterior probability, so as to determine whether a node is a malicious node, but when the number of active nodes in the network is insufficient or the number of packets processed by nodes is small, the false detection rate is high. Aiming at the problem that the existing malicious node detection methods in wireless sensor networks cannot be guaranteed by fairness and traceability of detection process, She et al. [25] present a blockchain trust model (BTM) for malicious node detection in wireless sensor networks. In BTM, through 3D space, it is realized by using blockchain intelligent contract and WSN quadrilateral measurement for localization of the detection of malicious nodes, and the consensus results of voting are recorded in the blockchain distributed. The model can effectively detect malicious nodes in WSNs and ensure the traceability of the detection process, but the consensus method in the model is the traditional POW workload proof method, which requires relatively large computational power and high energy consumption, so it is not especially suitable for the running environment of wireless sensor networks.

Li et al. [26] proposed a distributed and randomized detection algorithm to locate the attackers who inject polluted packets (IPAs). In this scheme, each node i maintains a set of suspicious nodes. In the beginning, all the neighbors of node i are added to a set of suspicious nodes; if the packets sent by its neighbor nodes are invalid, then the neighbor nodes that send valid packets are deleted from the suspicious nodes set; after n rounds of detection, the nodes in the set of suspicious nodes are malicious neighbors. Although the scheme can effectively detect malicious nodes in the network, it needs n rounds of detection, which will greatly increase the network communication overhead.

To sum up, all kinds of current research schemes have their own characteristics (Table 1). Comparison of advantages and disadvantages of each scheme makes a comparative analysis of relevant work. The detection schemes [14–16] based on multihop acknowledgment need to transmit a large number of acknowledgment packets, which will lead to high communication overhead. The detection schemes [17–22] based on trust evaluation need more monitoring nodes, which greatly increases the overhead of the network. And the current detection schemes of malicious nodes mainly focus on how to detect and locate malicious nodes in a single path. The HFDLMN scheme proposed in this paper does not need any complex evaluation model to evaluate and calculate the trust value of the node, nor any monitoring nodes, and the HFDLMN scheme can detect and locate malicious nodes in multiple paths.

3. Preliminaries and System Model

3.1. Preliminaries

Homomorphic Fingerprinting. Hendricks et al. first proposed homomorphic fingerprinting in [27]. The fingerprinting functions of homomorphic fingerprinting belong to a family of universal hash functions also. Let IF_{q^ω} denote a field of order q^ω, let K be the set of fingerprinting key, and let P_{q^ω}: $K \longrightarrow IF_{q^\omega}[x]$ be a deterministic algorithm that outputs monic irreducible polynomials of prime degree γ with coefficients in IF_{q^ω}; the polynomials are chosen with probabilities taken over the choice of input $r \in K$ uniformly at random; then a fingerprinting function $fp(r, d)$: $K \times IF_{q^\omega}^{\delta} \longrightarrow IF_{q^\omega}^{\gamma}$ can be defined as

Table 1: Comparison of advantages and disadvantages of each scheme.

Schemes	Advantages	Disadvantages
TWOACK [14]	Can be easily added to source routing protocols such as the DSR protocol	Greatly increases conflict and collision of network messages
CHEMAS [15]	Reduce the conflict and collision of network messages	Cannot resist the collusive attacks
PHACK [16]	Not only can detect a selective forwarding attack but also can recover the routing from the location at which the data were dropped	The scheme produces more ACK packets, which will consume more energy
MNDREL [17]	By analyzing the reputation value for the parent node evaluated by the child node, the malicious nodes in the network are effectively identified	The real-time performance of the deletion model has to be improved
GRFSN [18]	This scheme only needs to determine a trust threshold	Need complex evaluation model and the trust threshold is static, and the misjudgment rate of normal nodes being judged as malicious nodes is high
DNSMTM [19]	Can effectively detect malicious nodes and, with a higher detection rate, reduce the energy consumption of nodes	Need a complex evaluation model and the responsibility of monitoring nodes is too heavy
MSGSFS [20]	Can effectively alleviate selective forwarding attack and has less energy consumption	Need a complex game model and it can only solve the malicious packet loss behavior of single-hop nodes
ITEMBB [21]	The model overcomes the limitations of subjective weight allocation	Need complex evaluation model and the problem of static reputation value has not been solved
SMCSF [22]	Combine the neighbor node monitoring and watchdog mechanism	The responsibility of monitoring nodes is also heavy
IDSBS [23]	The collected information and its treatment are performed in a distributed way	The false detection rate of the system is high in the initial stage
MCMND [24]	Use multivariate classification to classify nodes	The false detection rate is high
BTM [25]	Using a blockchain intelligent contract, which can ensure the traceability of the detection process	The consensus method in the model is the traditional POW workload proof method, which requires relatively large computational power and high energy consumption
IPA [26]	Each node maintains a suspicious node set	Need n rounds of detection, which will greatly increase the network communication overhead

$fp(r, d(x))$: $p(x) \leftarrow P_{q^\omega}(r)$; return $(d(x) \bmod \cdot\, p(x))$. A fingerprinting function $fp(r, d)$: $K \times IF_{q^\omega}^{\delta} \longrightarrow IF_{q^\omega}^{\gamma}$ is homomorphic if $fp(r, d) + fp(r, d') = fp(r, d + d')$ and $b \cdot fp(r, d) = fp(r, b \cdot d)$ for any $r \in K$ and $d, d' \in IF_{q^\omega}^{\gamma}, b \in IF_{q^\omega}$. Let (encode, decode) be a linear erasure code with coefficients $b_{ij} \in IF_{q^\omega}$, for $i \in [1, n]$ and $j \in [1, m]$; if $d_1, \ldots, d_n \leftarrow \text{encode}^{\delta}(B)$, then for a homomorphic fingerprinting function $fp(r, d)$: $K \times IF_{q^\omega}^{\delta} \longrightarrow IF_{q^\omega}^{\gamma}$, the following equation holds: $fp(r, d_i) = \text{encode}_i^{\gamma}(fp(r, d_1), \ldots, fp(r, d_m))$, where $r \in K$ and $i \in [1, n]$.

3.2. Network Model. The sensor network is composed of ordinary nodes, malicious nodes, and base station (BS). Before deployment, each node i is assigned a unique identity ID_i, a random number r, $r \in IF_{2^\omega}$, and a symmetric key $K_{i,\text{BS}}$ shared with a base station. After the network is deployed to the target area, all nodes do not move. Adopting the method of [13], each node establishes multiple disjoint paths with the base station, and each node sends the data to the base station through multiple paths, for example. In Figure 1, the source node D and the base station have established n data transmission paths. Assuming that node D wants to send the data to the base station, it first divides the data into n fragments that are different from each other and encodes the n fragments to n new fragments; then, using homomorphic fingerprint technology, n packets are generated and sent to the base station along n different paths, respectively. When the base station receives n packets, if all the packets are valid, it will recover the original data.

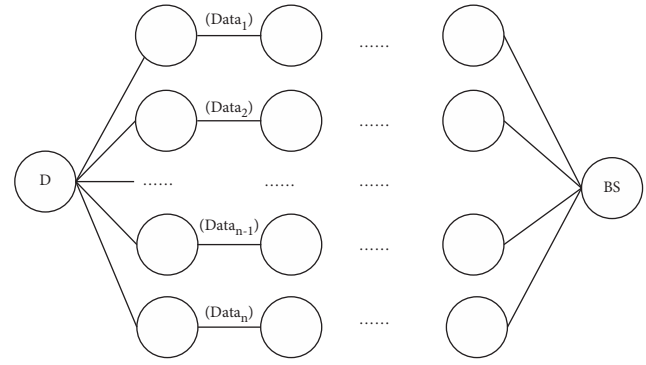

Figure 1: Source node D transmits data to a base station through multiple paths.

3.3. Attack Model and Security Goal. The HFDLMN scheme assumes that any intermediate forwarding nodes can be captured as malicious nodes by the attackers. These malicious nodes can launch pollution attacks by injecting false data into the network, forging or modifying the packets. The HFDLMN scheme does not consider other attacks such as

selective forwarding attacks but only considers pollution attacks. When the malicious node in the path receives the data, it will forge or modify the data with probability q and then forward it to the next-hop node. In the HFDLMN scheme, it is assumed that the calculation, storage, and communication capabilities of the base station are not limited, and the attackers can only capture ordinary sensor nodes, but not the base station.

Nowadays, there are several WSN standards (e.g., IEEE 802.15.4) that use different security levels at each layer. For instance, the network part of a packet is signed and encrypted with a network key and a data link layer with a DLL key. When an intermediate node receives a packet to retransmit, the DLL part needs to be verified; if it is not signed, the intermediate node drops the packet. Although the signature and encryption method can verify whether the packet has been modified, it cannot locate the malicious node that modifies the packet. The security goal of the HFDLMN scheme is not only to verify whether the packet is polluted but also to detect and locate the malicious nodes that launch pollution attacks.

4. Homomorphic Fingerprinting-Based Detection and Location of Malicious Nodes

The HFDLMN scheme proposed in this paper is divided into five steps: the source node generates the packets, the intermediate node forwards the packet, the base station detects the path of pollution attack, the base station locates the malicious node or malicious link, and base station recovers the original data.

4.1. Generating the Packets

4.1.1. Generating Data Segmentation. If the source node wants to send the data to the base station, it first divides the data into n fragments that are different from each other, namely, data $= < f_1, \ldots, f_n >$.

4.1.2. Coding Data Segmentation. Then, the source node generates n linearly independent vectors, the elements of the vectors are randomly picked from the field IF_{2^ω}, and the vector is denoted as $[g_{i,1}, \ldots, g_{i,n}]$, $i = 1, \ldots, n$. According to the following equation, the source node can get n new fragments, which are denoted as Y_i, $i = 1, \ldots, n$.

$$\begin{bmatrix} g_{1,1} & \cdots & g_{1,n} \\ \vdots & \ddots & \vdots \\ g_{n,1} & \cdots & g_{n,n} \end{bmatrix} \times \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix}. \tag{1}$$

4.1.3. Generating and Sending the Packets. After coding data segmentation, the source node generates n packets, which are denoted as $\text{Packet}_i = < \text{Seq\_Number}_k, hp(r, f_i), G_i, Y_i >, i = 1, \ldots, n$, where the n packets have the same number which is denoted by Seq_Number_k, $hp(r, f_i)$ denotes the fingerprinting of f_i, which is computed by fingerprinting function $hp(.)$ and the random number r, $r \in IF_{2^\omega}$, $G_i = [g_{i,1}, \ldots, g_{i,n}]$ denotes the coding vector of f_i, and Y_i denotes the new data fragment after coding; then, the Packet_i is sent to the base station along n different paths, respectively.

4.2. Forwarding the Packets.

All intermediate forwarding nodes maintain a data forwarding table (DFT), which is shown in Table 2, where the Seq_Number field stores the packet number, and the Finger_printing field stores the fingerprinting of f_i, the Encoding_Vector field stores the coding vector of f_i, the Encoded_DataBlock field stores the new data fragment after coding, and the DFT only stores the packets that were forwarded the last three times. When the intermediate forwarding node j receives $\text{Packet}_i = < \text{Seq\_Number}_k, hp(r, f_i), G_i, Y_i >$, it will delete the first record stored in the DFT and store the currently received packet in the DFT, which makes the DFT only store the packets that were forwarded the last three times and facilitate the base station query, and then it forwards the $Packet_i$ to the next intermediate forwarding node.

4.3. Detecting the Path of Pollution Attack.

After the base station receives n packets with the same number from n paths, it first gets $hp(r, f_i)$ Y_i and G_i from the $\text{Packet}_i = < \text{Seq\_Number}_k, hp(r, f_i), G_i, Y_i >, i = 1, \ldots, n$ and computes $hp(r, Y_i), i = 1, \ldots, n$ respectively; then, it randomly picks $t_1, \ldots, t_n$ from the field IF_{2^ω} and constructs a new vector $V = [v_1, \ldots, v_n]$ according to

$$[v_1, \ldots, v_n] = [t_1, \ldots, t_n] \begin{bmatrix} G_1 \\ \ldots \\ G_n \end{bmatrix} = \left[\sum_{i=1}^{n} t_i g_{i1}, \ldots, \sum_{i=1}^{n} t_i g_{in} \right], \tag{2}$$

$$v_1 hp(r, f_1) + \cdots + v_n hp(r, f_n) = t_1 hp(r, Y_1) + \cdots + t_n hp(r, Y_n). \tag{3}$$

The base station can validate the validity of n packets according to equation (3). If equation (3) holds, all the n packets are valid, and it will be performed in Section 4.5 to recover the original data; otherwise, it means that malicious nodes polluted the packets in one path or more paths. Then, the base station can detect which packet is polluted according to equation (4). If equation (4) does not hold, the Packet_i is polluted, and there are malicious nodes in the path; the base station will execute Algorithm 1 in Section 4.4 to detect and locate the malicious node.

$$hp(r, Y_i) = G_i \times [hp(r, f_1), \ldots, hp(r, f_n)]^T, \quad i = 1, \ldots, n. \tag{4}$$

4.4. Locating the Malicious Node. When the base station finds the path of pollution attack and the pollution packet $\text{Packet}_m = <\text{Seq\_Number}_k, hp(r, f_m), G_m, Y_m>$, it assumes that there are m hops in the attack path from the source node to the base station, it is represented by $(n_1, n_2, \ldots, n_m, \text{BS})$, where n_1 represents the source node, and the remaining nodes represent intermediate forwarding nodes in the path. In order to locate the malicious nodes in the path, from the source node, the base station first informs each node to send the response packet to the base station along the attack path in turn, and the response packet p_i is generated according to equation (5). In equation (5), ID_i represents the identity of node n_i, and $\text{SPacket}_i = <n_i.DF\ T.\text{Seq\_Number}_k \| n_i.DF\ T.hp(r, f_i) \| n_i.DF\ T.G_i \| n_i.DF\ T.Y_i>$, p_{i-1} represents the response packet sent by the previous hop node n_{i-1}, Timestamp represents the timestamp, and $\|$ represents the connection operation.

$$p_i = E_{K_{i,\text{BS}}}\left(ID_i \| \text{SPacket}_i \| p_{i-1} | \text{Timestamp}\right). \tag{5}$$

After the base station receives the response packet p_m, it will execute Algorithm 1 to locate the malicious node or the malicious link. From back to front, first, it sequentially decrypts p_i with the symmetric key $K_{i,\text{BS}}$ shared with the node n_i and gets the ID_i and SPacket_i ; then, from the node n_1, the base station compares SPacket_i with the pollution packet Packet_m received by the base station in turn; if it is equal, it means that node n_i is a malicious node or the link between node n_i and node n_{i-1} is a malicious link.

4.5. Recovering the Original Data. After the base station receives n linearly independent packets with the same number from n paths, it can validate the validity of n packets according to equation (3); if equation (3) holds, all the n packets are valid, and it will recover the original data. It first gets Y_i and G_i from the $\text{Packet}_i = <\text{Seq\_Number}_k, hp(r, f_i), G_i, Y_i>, i = 1, \ldots, n$, and generates the vector coefficient matrix T, as shown in equation (6). Because n vectors G_i are vector linearly independent, the coefficient matrix T is full rank, and the base station can get the inverse matrix T^{-1} and recover the original Data $= <f_1, \ldots, f_n>$ according to equation (7).

$$T = \begin{bmatrix} g_{1,1} & \cdots & g_{1,n} \\ \vdots & \ddots & \vdots \\ g_{n,1} & \cdots & g_{n,n} \end{bmatrix}, \tag{6}$$

$$\begin{bmatrix} f_1 \\ \vdots \\ f_2 \end{bmatrix} = T^{-1} \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix}. \tag{7}$$

5. Proof and Analysis of Related Theorems

5.1. Proof of Malicious Path Detectability. The base station can validate the validity of n packets according to equation (3); if equation (3) holds, all the n packets are valid, and it will recover the original data; otherwise, the base station can detect which packet is polluted according to equation (4); if equation (4) does not hold, the Packet_i is polluted, and there are malicious nodes in the path of sending the pollution packet. This section will prove the correctness of equations (3) and (4).

Theorem 1. *After the base station receives n linearly independent packets with the same number from n paths, if all the packets it receives are valid, then equation (3) holds; if there are t (t < n) packets that are polluted, then equation (3) does not hold.*

Proof

(1) If all the packets received by the base station are valid, then

$$\begin{aligned} v_1 hp(r, f_1) + \cdots + v_n hp(r, f_n) &= hp(r, v_1 f_1) + \cdots + hp(r, v_n f_n) \\ &= hp\left(r, [v_1, \ldots, v_n][f_1, \ldots, f_n]^T\right) \\ &= hp\left(r, [t_1, \ldots, t_n] \begin{bmatrix} G_1 \\ \cdots \\ G_n \end{bmatrix} [f_1, \ldots, f_n]^T\right) \\ &= hp\left(r, [t_1, \ldots, t_n] \begin{bmatrix} g_{1,1} & \cdots & g_{1,n} \\ \vdots & \ddots & \vdots \\ g_{n,1} & \cdots & g_{n,n} \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix}\right) \\ &= hp\left(r, [t_1, \ldots, t_n] \begin{bmatrix} Y_1 \\ \cdots \\ Y_n \end{bmatrix}\right) \\ &= hp(r, t_1 Y_1 + \cdots + t_n Y_n) \\ &= t_1 hp(r, Y_1) + \cdots + t_n hp(r, Y_n) \\ &\quad hp(r, v_1 f_1) + \cdots + hp(r, v_n f_n). \end{aligned} \tag{8}$$

That is, Eq. (3) holds.

(2) Assuming that one of the n packets received by the base station is polluted by the forwarding malicious node n_i in the path, and the polluted packet is $Packet_i' = < Seq\_Number_k, hp(r, f_i^p), G_i^p, Y_i^p >$, where f_i^p, G_i^p, Y_i^p are false data injected by the malicious node n_i, $hp(r, f_i^p)$ denotes the fingerprinting of f_i^p, computed by fingerprinting function $hp(.)$ and the random number r, $r \in IF_{2^\omega}$, then,

$$\begin{aligned}
v_1 hp(r, f_1) + \cdots + v_i hp\left(r, f_i^p\right) + \cdots + v_n hp(r, f_n) &= hp(r, v_1 f_1) + \cdots + hp\left(r, v_i f_i^p\right) + \cdots + hp(r, v_n f_n) \\
&= hp\left(r, [v_1, \ldots, v_i, \ldots, v_n]\left[f_1, \ldots, f_i^p, \ldots, f_n\right]^T\right) \\
&= hp\left(r, [t_1, \ldots, t_n]\begin{bmatrix} G_1 \\ \cdots \\ G_i^p \\ \cdots \\ G_n \end{bmatrix}\left[f_1, \ldots, f_i^p, \ldots, f_n\right]^T\right) \\
&= hp\left(r, [t_1, \ldots, t_n]\begin{bmatrix} g_{1,1} & \cdots & g_{1,n} \\ \cdots & \cdots & \cdots \\ g_{i,1}^p & \cdots & g_{i,n}^p \\ \cdots & \cdots & \cdots \\ g_{n,1} & \cdots & g_{n,n} \end{bmatrix}\begin{bmatrix} f_1 \\ \cdots \\ f_i^p \\ \cdots \\ f_n \end{bmatrix}\right) \\
&= hp\left(r, [t_1, \ldots, t_n]\begin{bmatrix} Y_1^p \\ \cdots \\ Y_i^p \\ \cdots \\ Y_n^p \end{bmatrix}\right) \\
&= hp\left(r, t_1 Y_1^p + \cdots + t_i Y_i^p + \cdots + t_n Y_n^p\right) \\
&= t_1 hp\left(r, Y_1^p\right) + \cdots + t_i hp\left(r, Y_i^p\right) + \cdots + t_n hp(r, Y_n^p) \\
&\quad hp(r, v_1 f_1) + \cdots + hp\left(r, v_i f_i^p\right) + \cdots + hp(r, v_n f_n).
\end{aligned} \tag{9}$$

Because the malicious node n_i has no $\langle f_1, \ldots, f_{i-1}, f_i + 1, \ldots, f_n \rangle$ and $\langle G_1, \ldots, G_{i-1}, G_{i+1}, \ldots, G_n \rangle$, it cannot construct $\langle Y_1^p, \ldots, Y_n^p \rangle$ and make $\langle Y_1^p == Y_1, \ldots, Y_n^p == Y_n \rangle$; as a result, $t_1 hp(r, Y_1^p) + \cdots + t_i hp(r, Y_i^p) + \cdots + t_n hp(r, Y_n^p) \neq v_1 hp(r, f_1) + \cdots + v_i hp(r, f_i^p) + \cdots + v_n hp(r, f_n)$.

So, without considering the link error of the network, if one or more packets are polluted, then Eq. (3) does not hold.

Theorem 1 is proved. □

Theorem 2. *After the base station receives n linearly independent packets with the same number from n paths, if all the packets received by the base station are valid, then equation (4) holds; otherwise, equation (4) does not hold, and the path of sending pollution packet is the malicious path.*

Proof

(1) Assuming that $\text{Packet}_i = < \text{Seq\_Number}_k, hp(r, f_i), G_i, Y_i >$ is valid, then,

$$\begin{aligned}
G_i \times [hp(r, f_1), \ldots, hp(r, f_n)]^T &= \left[g_{i,1}, \ldots, g_{i,n}\right] \times [hp(r, f_1), \ldots, hp(r, f_n)]^T \\
&= g_{i,1} hp(r, f_1) + \cdots + g_{i,n} hp(r, f_n) \\
&= hp\left(r, g_{i,1} f_1\right) + \cdots + hp\left(r, g_{i,n} f_n\right) \\
&= hp\left(r, g_{i,1} f_1 + \cdots + g_{i,n} f_n\right) \\
&= hp\left(r, \left[g_{i,1}, \ldots, g_{i,n}\right]\begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix}\right) \\
&= hp(r, Y_i).
\end{aligned} \tag{10}$$

That is, if the packet has not been modified, $G_i \times [hp(r, f_1), \ldots, hp(r, f_n)]^T = hp(r, Y_i)$, therefore, (4) holds.

(2) Assuming that one of the n packets received by the base station is polluted by the forwarding malicious node n_i in the path, and the polluted packet is $\text{Packet}_i' = <\text{Seq\_Number}_k, hp(r, f_i^p), G_i^p, Y_i^p>$, where f_i^p, G_i^p, Y_i^p are false data injected by the malicious node n_i, $hp(r, f_i^p)$ denotes the fingerprinting of f_i^p, computed by fingerprinting function $hp(.)$ and the random number r, $r \in IF_{2^\omega}$, then

$$\begin{aligned}
&G_i^p \times \left[hp\left(r, f_1\right), \ldots, hp\left(r, f_i^p\right), \ldots, hp\left(r, f_n\right)\right)\Big]^T \\
&= \left[g_{i,1}^p, \ldots, g_{i,i}^p, \ldots, g_{i,n}^p\right] \times \left[hp\left(r, f_1\right), \ldots, hp\left(r, f_i^p\right), \ldots, hp\left(r, f_n\right)\right)\Big]^T \\
&= g_{i,1}^p hp\left(r, f_1\right) + \cdots + g_{i,i}^p hp\left(r, f_i^p\right) + \cdots + g_{i,n}^p hp\left(r, f_n\right) \\
&= hp\left(r, g_{i,1}^p f_1\right) + \cdots + hp\left(r, g_{i,i}^p f_i^p\right) + \cdots + hp\left(r, g_{i,n}^p f_n\right) \\
&= hp\left(r, g_{i,1}^p f_1 + \cdots + g_{i,i}^p f_i^p + \cdots + g_{i,n}^p f_n\right) \\
&= hp\left(r, \left[g_{i,1}^p, \ldots, g_{i,i}^p, \ldots, g_{i,n}^p\right] \begin{bmatrix} f_1 \\ \cdots \\ f_i^p \\ \cdots \\ f_n \end{bmatrix}\right) \\
&= hp\left(r, Y_i^{p'}\right).
\end{aligned} \tag{11}$$

Because the malicious node n_i has no $\langle f_1, \ldots, f_{i-1}, f_{i+1}, \ldots, f_n\rangle$, it cannot construct $Y_i^{p'}$ and make $Y_i^{p'} == Y_i^p$; as a result, $hp(r, Y_i^{p'}) \neq hp(r, Y_i^p)$.

So, without considering the link error of the network, if one or more packets are polluted, then (4) does not hold, and the path to send the pollution packet is the malicious path.

Theorem 2 is proved. □

5.2. Probability Analysis of Legitimate Nodes Being Misjudged as Malicious Nodes. Because of the link error, a legitimate node will be misjudged as a malicious node by forwarding a packet distorted by the link error. This section will analyze the probability of the legitimate node being misjudged as a malicious node in the case of a link error.

Theorem 3. *Assuming that the probability of link error is q, and the number of packets transmitted in the path is S in time period T, the probability of legitimate nodes being misjudged as malicious nodes is*

$$P_m = \sum_{k=1}^{S} \binom{S}{k} q^k (1-q)^{S-k}. \tag{12}$$

Proof. Let X be the times that misjudged packets are detected, it is obvious that X satisfies the binomial distribution of parameters S and q, that is, X-b (S, q), and the distribution law of X is as follows:

$$P\{X = k\} = \binom{S}{k} q^k (1-q)^{S-k}, \quad k = 0, 1, \ldots, S. \tag{13}$$

Therefore, the probability of legitimate nodes being misjudged as malicious nodes is

$$\begin{aligned}
P_m &= P\{X \geq 1\} \\
&= \sum_{k=1}^{S} \binom{S}{k} q^k (1-q)^{S-k}.
\end{aligned} \tag{14}$$

So, Theorem 3 is proved. □

5.3. Time Complexity Analysis of the Algorithm. If there are malicious nodes in the path, the base station will execute Algorithm 1 in Section 4.4 to detect and locate the malicious node or malicious link, this section will analyze the time complexity of Algorithm 1.

Basic operations of Algorithm 1 are to get the $SPacket_i$ from the response packet and the comparison of $SPacket_i$ with the pollution packet Packet_m. Getting the $SPacket_i$ from the response is mainly to perform m cyclic operations; therefore, the time complexity of basic operation of getting the $SPacket_i$ from the response is $O(m)$. The comparison of $SPacket_i$ with the pollution packet Packet_m is to search whether the pollution packet Packet_m is in the array $SPacket$; therefore, the time complexity of basic operation of the comparison of $SPacket_i$ with the pollution packet Packet_m is $O(m)$, too. So, the time complexity of the two basic operations is $O(2m)$; that is, the time complexity of Algorithm 1 is $O(n)$.

Table 2: Data forwarding table (DFT).

Seq_Number	Finger_printing	Encoding_Vector	Encoded_DataBlock
.....	.....	.....	.....
Seq-Number$_k$	$hp(r, Y_i)$	G_i	Y_i
.....	.....	.....	.....

6. Security Analysis

In the HFDLMN scheme, the malicious nodes not only can launch pollution attacks by injecting false data into the network, forging or modifying the packets, but also can also modify or delete the response packet p_i sent by its previous node, so that the malicious nodes can avoid or interfere with the base station to perform the detection of the malicious nodes. Because the pollution attack launched by a malicious node can be detected by (3) and (4), and if the malicious node normally forwards the[[parms resize(1),pos(50,50),size(200,200),bgcol(156)]] scheme resists to the malicious nodes avoiding or interfering with the base station to perform the detection of the malicious nodes.

Theorem 4. *Any malicious node can be detected after modifying, deleting, or not sending response packets.*

Proof. In order to interfere with the detection of malicious nodes performed by the base station, when the base station informs each node to send the response packet p_i to the base station along the path, in turn, the malicious node n_i can perform the following operations: (a) Attempt to modify the data of the response packet p_{i-1} sent by its previous node; however, the response packet p_{i-1} is a key chain generated by encrypting the time stamp, p_{i-2}, the identity of the node n_{i-1}, and SPacket$_{i-1}$ with the symmetric key $K_{i-1,\mathrm{BS}}$ shared by node n_{i-1} and base station; that is, $p_{i-1} = E_{K_{i-1,BS}}(ID_{i-1} \| \mathrm{SPacket}_{i-1} || p_{i-2} || \mathrm{Timestamp})$, because the malicious node n_i does not have the symmetric key $K_{i-1,\mathrm{BS}}$ shared by the node n_{i-1} and the base station, and it cannot modify the data in the response packet p_{i-1} sent by its previous node. (b) Try to delete the data in the response packet p_{i-1} sent by its previous node, also because the malicious node n_i does not have the symmetric key $K_{i-1,\mathrm{BS}}$ shared by the node n_{i-1} and the base station, so it cannot delete the data in the response packet p_{i-1} sent by its previous node. (c) Try not to send its own query response packet to the base station, that is, directly forwards the received response packet p_{i-1} from its previous node to its next node n_{i+1}. In this case, according to Algorithm 1, when the base station tries to decrypt the response packet p_i with the symmetric key $K_{i,\mathrm{BS}}$ shared with the malicious node n_i, the malicious node did not send its own response packet, so the base station cannot correctly decrypt the response packet p_i; therefore, it can be determined that the node n_i is a malicious node. (d) Try to send false data to the base station and interfere with the detection of malicious nodes. For example, the malicious node sends unmodified data to the base station; according to Algorithm 1, the base station can correctly locate the malicious link composed of the malicious node and its next-hop node; similarly, the malicious node can also send a modified data to the base station; according to Algorithm 1, the base station can correctly locate the malicious link composed of the malicious node and its next-hop node.

In summary, in the HFDLMN scheme, any malicious node can be detected after modifying, deleting, or not sending response packets.

So, Theorem 4 is proved. □

7. Simulation

In this paper, the performance of the HFDLMN scheme is evaluated from the aspects of the detection probability of malicious path, the location probability of malicious node, and the false detection probability of normal nodes and paths. The simulation experiment environment is carried out on OMNeT++ platform, with 100 nodes randomly distributed in a square area of 400m × 400 m, each node is assigned a unique *ID*, the nodes will not move after deployment, and the base station is deployed in the center of the area. By adjusting the communication range of each node, each node has at least four neighbor nodes, and each node establishes four disjoint paths to the base station. Some nodes in the network are randomly selected as data source nodes and malicious nodes, and others as intermediate forwarding nodes. The source node sends the packet to the base station by multihop every 1 second, and the length of each packet is 256 bytes. The initial energy of each node is 1J, and the energy consumption of transmission and receiving is 50 nJ/bit. When a malicious node becomes an intermediate forwarding node, it will forge or modify packets with a probability from 0.1 to 0.7. For each set parameter, the average value obtained by 100 simulations is taken. The parameter settings of the experimental simulation are shown in Table 3.

Figure 2 describes the detection probability of a malicious path when the malicious node forges or modifies packets with a probability of 0.1, 0.3, 0.5, and 0.7, and there is a malicious node in one of the four paths from the source node to the base station. From Figure 2, it can be seen that the number of packets that need to be sent to successfully detect malicious paths is related to the probability q of malicious node modifying data. The higher the probability of malicious node modifying data, the less packets need to be sent to successfully detect the malicious path; for example, when the probability q of malicious node modifying data is 0.3, in order to detect the malicious path successfully, the source node needs to send 14 packets; when the probability q of malicious nodes modifying data is 0.5, the base station can successfully detect the malicious path by only sending 9 packets.

Figure 3 describes the detection probability of a malicious path when the malicious node forges or modifies

Input: the path of pollution attack , the pollution packet $\mathbf{Packet_m}$, and the response packet $\mathbf{p_m}$
Output: malicious node or malicious link
For $(i = m;\ i > 1;$ i--) do
  $\mathbf{p_i'} = \mathbf{D_{K_{iBS}}}(\mathbf{p_i})$
  If cannot get accurately $\mathbf{p_i'}$ then
    Return $\mathbf{ID_{i-1}}$
  End if
  $\mathbf{SPacket}[\mathbf{i}] = \mathbf{SPacket_i}$
  $\mathbf{ID}[\mathbf{i}] = \mathbf{ID_i}$
End for
For$(i = 1;\ i \le m;\ i{+}{+})$ do
  If $\mathbf{SPacket}[\mathbf{i}] == \mathbf{Packet_m}$ then
    Return $\mathbf{ID_{i-1}}$ and $\mathbf{ID_i}$
  End if
End for

Algorithm 1: Location of malicious nodes.

Table 3: Experimental simulation parameters.

Parameter	Value or range
Network deployment area (m)	400×400
Number of nodes in the network	100
Initial energy of each node (J)	1
The energy consumption of the transmission and receiving (nJ/bit)	50
Time interval of sending packet (S)	1
The number of malicious nodes	20
The probability of malicious nodes modifying packets	0.1–0.7
The probability of link error	0.005–0.06
The simulation time (S)	600

packets with a probability of 0.1, 0.3, and 0.5, and the number of paths with malicious nodes is 2 and 3. From Figure 3, it can be seen that the number of packets that need to be sent to successfully detect malicious paths is not only related to the probability q of malicious node modifying data but also related to the number of paths with malicious nodes. The higher the probability of malicious node modifying data and the more number of paths with malicious nodes, the less packets need to be sent to successfully detect the malicious path; for example, when the probability q of malicious node modifying data is 0.1 and the number of paths with malicious nodes is 2, in order to detect the malicious path successfully, the source node needs to send 8 packets; when the probability q of malicious node modifying data is 0.3 and the number of paths with malicious nodes is 3, the base station can successfully detect the malicious path by only sending 4 packets.

Figure 4 describes the locating probability of the malicious node when the malicious node forges or modifies packets with a probability of 0.1 and 0.3, and the number of paths with malicious nodes is 1, 2, and 3. From Figure 4, it can be seen that with the probability increase of the malicious node modifying data and the number increase of malicious paths, the probability of successfully locating malicious nodes will increase; for example, when there is a malicious node in only one path and the probability q of

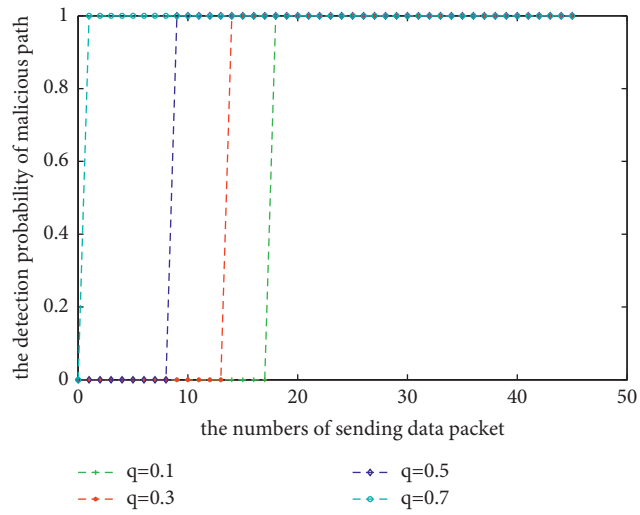

Figure 2: The detection probability of a malicious path when there is a malicious node in only one path.

malicious node modifying data is 0.1, the source node sends 15 packets and the probability of successfully locating the malicious node is about 84%; when there are malicious nodes in two paths and the probability q of malicious nodes modifying data is 0.3, the source node only sends 10 packets

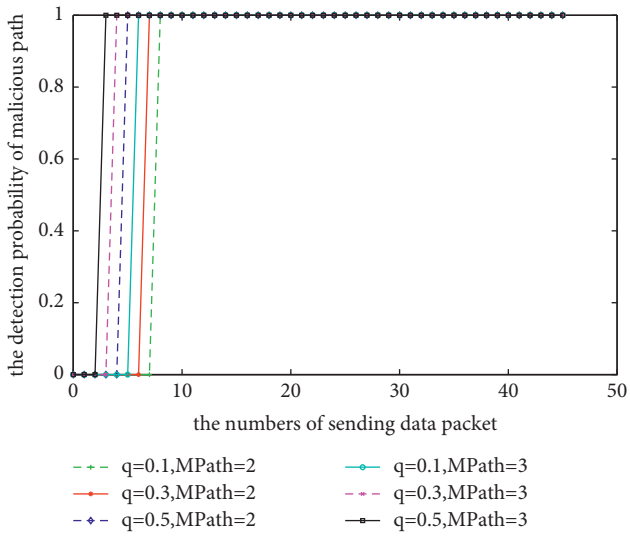

Figure 3: The detection probability of a malicious path when there are malicious nodes in multiple paths.

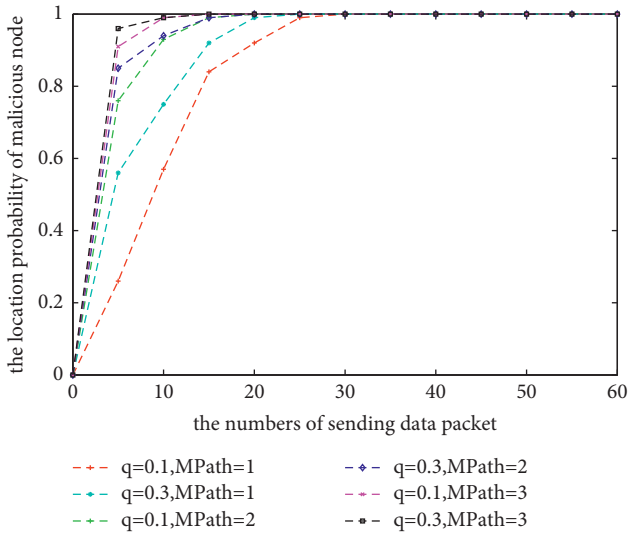

Figure 4: The locating probability of the malicious node.

and the probability of successfully locating the malicious node is about 94%.

Because of the link error, a legitimate node will forward a packet distorted by the link error, so that the legitimate node and path are misjudged as malicious node and path. Figure 5 describes the probability of the legitimate node and path being misjudged as malicious node and path when the probability of link error is from 0.005 to 0.06 and the number of packets transmitted in the path is 100 in a certain period of time. From Figure 5, it can be seen that with the probability increase of the link error, the probability of the legitimate node and path being misjudged as malicious node and path will increase; for example, when the probability of link error is 0.01, the probability of the legitimate node and path being misjudged as malicious node and path is about 5%, and when the probability of link error is 0.05, the false detection probability of the legitimate node and path is about 19%.

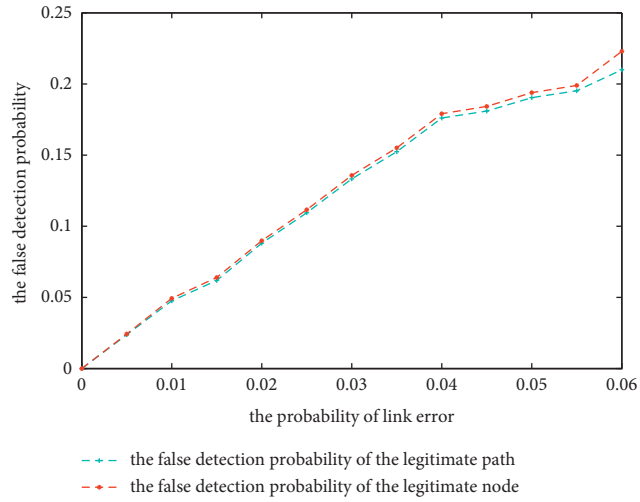

Figure 5: The false detection probability of legitimate nodes and paths.

8. Conclusions

To detect and locate malicious nodes in multiple paths, this paper presents a malicious node detection and location scheme based on homomorphic fingerprint and coding technology in wireless sensor networks, HFDLMN. In the HFDLMN scheme, the source node generates n packets and sends them to the base station along n paths, respectively; the base station determines whether there are malicious nodes in each path by verifying the validity of the packets; if there are malicious nodes in one or some paths, the location algorithm of a malicious node is implemented to locate the specific malicious nodes in the path. The HFDLMN scheme does not need any complex evaluation model to evaluate and calculate the trust value of the node, nor any monitoring nodes. Using a key chain, the HFDLMN scheme can resist malicious nodes to avoid or interfere with the base station to detect malicious nodes. Theoretical analysis results show that the HFDLMN scheme is secure and effective, the simulation results demonstrate that the HFDLMN scheme can effectively detect malicious paths and malicious nodes, with a higher detection rate; for example, if there are malicious nodes in two paths and the probability q of malicious nodes modifying data is 0.3, the source node only sends 10 packets and the probability of successfully locating the malicious node is about 94%. In the future, we aim to extend this work into designing a new detection and location of malicious nodes scheme among Internet of Things devices.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this study.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62002143 and in part by the Natural Science Foundation of Jiangxi Province under Grant 20192BAB217007.

References

[1] E. Nurellari, D. Mclernon, M. Ghogho, and S. Aldalahmeh, "Distributed binary event detection under data-falsification and energy-bandwidth limitation," *IEEE Sensors Journal*, vol. 16, no. 16, pp. 6298–6309, 2016.

[2] A. Alanezi, B. Abd-El-Atty, H. Kolivand et al., "Securing digital images through simple permutation-substitution mechanism in cloud-based smart city environment," *Security and Communication Networks*, vol. 2021, Article ID 6615512, 17 pages, 2021.

[3] U. Baroudi and M. E. Haque, "Ambient self-powered cluster-based wireless sensor networks for industry 4.0 applications," *Soft Computing*, vol. 25, no. 4, pp. 1859–1884, 2021.

[4] R. Yadav, W. Zhang, I. A. Elgendy, G. Dong, and S. Prakash, "Smart healthcare: RL-based task offloading scheme for edge-enable sensor networks," *IEEE Sensors Journal*, vol. 2021, Article ID 3096245, 2021.

[5] A. Chaaf, M. S. A. Muthanna, A. Muthanna, S. Alhelaly, and A. A. A. El-Latif, "REVOHPR: relay-based void hole prevention and repair by virtual routing in clustered multi-AUV underwater wireless sensor network," *Security and Communication Networks*, vol. 2021, Article ID 9969605, 20 pages, 2021.

[6] J. Grover and S. Sharma, "Security issues in wireless sensor network—a review," in *Proceeding of the 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, pp. 397–404, Noida, India, September 2016.

[7] B. Zhu, V. Addada, S. Setia, S. Jajodia, and S. Roy, "Efficient distributed detection of node replication attacks in sensor networks," in *Proceeding of the 23rd Annual Computer Security Applications Conference (ACSAC '07)*, pp. 257–267, Miami Beach, FL, USA, December 2007.

[8] R. Rongxing Lu, X. Xiaodong Lin, H. Haojin Zhu, X. Xiaohui Liang, and X. Xuemin Shen, "BECAN: a bandwidth-efficient cooperative authentication scheme for filtering injected false data in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 1, pp. 32–43, 2012.

[9] A. Liu, Z. Zheng, C. Zhang, Z. Chen, and X. Shen, "Secure and energy-efficient disjoint multipath routing for WSNs," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 3255–3265, 2012.

[10] G. S. M. D'Souza and R. J. Varaprasad, "Digital signature-based secure node disjoint multipath routing protocol for wireless sensor networks," *Sensors Journal, IEEE*, vol. 12, no. 10, pp. 2941–2949, 2012.

[11] H. Xu, L. Huang, C. Qiao, Y. Zhang, and Q. Sun, "Bandwidth-power aware cooperative multipath routing for wireless multimedia sensor networks," *IEEE Transactions on Wireless Communications*, vol. 11, no. 4, pp. 1532–1543, 2012.

[12] R. Marjan, D. Behnam, A. B. Kamalrulnizam, A. R. Shukor, and N. M. Ali, "Interference-aware multipath routing protocol for QoS improvement in event-driven wireless sensor networks," *Tsinghua Science and Technology*, vol. 16, no. 5, pp. 475–490, 2011.

[13] S. Li and Z. Wu, "Node-disjoint parallel multi-path routing in wireless sensor networks," in *Proceedings of the 2nd International Conference on Embedded Software and Systems (ICESS'05)*, pp. 210–215, Xi'an, china, December 2005.

[14] K. Balakrishnan, J. Deng, and P. K. Varshney, "TWOACK: preventing selfishness in mobile ad hoc networks," in *Proceedings of the Wireless Communications & Networking Conference*, pp. 2137–2142, IEEE, New Orleans, LA, USA, April 2005.

[15] B. Xiao, B. Yu, and C. Gao, "CHEMAS: i," *Journal of Parallel and Distributed Computing*, vol. 67, no. 11, pp. 1218–1230, 2007.

[16] A. Liu, M. Dong, K. Ota, and J. Long, "PHACK: an efficient scheme for selective forwarding attack detection in WSNs," *Sensors*, vol. 15, no. 12, pp. 30942–30963, 2015.

[17] H. Yang, X. Zhang, and F. Cheng, "A novel algorithm for improving malicious node detection effect in wireless sensor networks," *Mobile Networks And Applications*, vol. 2020, Article ID s11036-019-01492-4, 2020.

[18] D. Xiao, J. Feng, and Q. Zhou, "Gauss reputation framework for sensor networks," *Journal on Communications*, vol. 29, no. 3, pp. 47–53, 2008.

[19] G. Zheng, B. Gong, and Y. Zhang, "Dynamic network security mechanism based on trust management in wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6667100, 10 pages, 2021.

[20] H. Liao and S. Ding, "Mixed and continuous strategy monitor-forward game based selective forwarding solution in WSN," *International Journal of Distributed Sensor Networks*, vol. 2015, no. 11, 13 pages, Article ID 359780, 2015.

[21] Z. Zhou and N. Shao, "An improved trust evaluation model based on Bayesian for WSNs," *Chinese Journal of Sensors and Actuators*, vol. 29, no. 6, pp. 927–933, 2016.

[22] H. Zhou, Y. Wu, L. Feng, and D. Liu, "A security mechanism for cluster-based WSN against selective forwarding," *Sensors*, vol. 16, no. 9, pp. 1537–1552, 2016.

[23] D. Silva, M. Martins, B. Rocha, A. Loureiro, L. B. Ruiz, and H. C. Wong, "Decentralized intrusion detection in wireless sensor networks," in *Proceedings of the 1st ACM international workshop on Quality of service*, pp. 16–22, Monteral, Canada, October 2005.

[24] H. Liu, J. Cui, and H. Dai Hongjun, "Multivariate classification-based malicious node detection for wireless sensor network," *Chinese Journal of Sensors and Actuators*, vol. 24, no. 5, pp. 771–777, 2011.

[25] W. She, Q. Liu, Z. Tian, J.-S. Chen, B. Wang, and W. Liu, "Blockchain trust model for malicious node detection in wireless sensor networks," *IEEE Access*, vol. 7, pp. 38947–38956, 2019.

[26] Y. Li and J. C. S. Lui, "Identifying pollution attackers in network-coding enabled wireless mesh networks," in *Proceedings of the 2011 20th International Conference on Computer Communications and Networks (ICCCN)*, pp. 1–6, Maui, HI, USA, August 2011.

[27] J. Hendricks, G. R. Ganger, and M. K. Reiter, "Verifying distributed erasure-coded data," in *Proceedings of 26th ACM Symposium on Principles of Distributed Computing*, pp. 1–8, Portland, OR, USA, August 2007.