RESEARCH ARTICLE OPEN ACCESS

WAT Calculation in EDTM

Anu Thomas
M Tech
Department of Computer Science & Engineering
Cochin Institute of Science and Technology
MG University
Kerala - India

ABSTRACT

Trust models have been recently suggested as an effective security mechanism for Wireless Sensor Networks (WSNs). However, most current research works only take communication behaviour into account to calculate sensor nodes' trust value, which is not enough for trust evaluation due to the widespread malicious attacks. In this paper, we propose an Efficient Distributed Trust Model (EDTM) for WSNs by considering security measure to attain privacy and authentication using RSA algorithm. First, according to the number of packets received by sensor nodes, direct trust and recommendation trust are selectively calculated. Then, communication trust, energy trust and data trust are considered during the calculation of direct trust. Furthermore, trust reliability and familiarity are defined to improve the accuracy of recommendation trust. The proposed EDTM can evaluate trustworthiness of sensor nodes more precisely and prevent the security breaches more effectively. Simulation results show that EDTM outperforms other similar models, e.g., NBBTE trust model. For providing more security here we use the technique RSA algorithm for privacy and authentication. The main 3 factors used in the communication than the trust model are Link quality, Distance between the nodes and the Security Measures.

Keywords:- Trust Model, Link quality, RSA Algorithms, Secure, Distance.

I. INTRODUCTION

WSNS are emerging technologies that have been widely used in many applications such as emergency response, health care monitoring, battlefield surveillance, habitat monitoring, traffic management, smart power grid, etc. However, the wireless and resource - constraint nature of a sensor network makes it an ideal medium for malicious attackers to intrude the system. Thus, providing security is extremely important for the safe application of WSNs.

Various security mechanisms, e.g., cryptography, authentication, confidentiality, and message integrity, have been proposed to avoid security threats such as eaves-dropping, message replay, and fabrication of messages. However, these approaches still suffer from many security vulnerabilities, such as node capture attacks and denial-of-service (DoS) attacks. The traditional security mechanisms can resist external attacks, but cannot solve internal attacks effectively which are used by the captured nodes. To establish secure communications, we need to ensure that all communicating nodes are trusted. This highlights the fact that it is critical to establish a trust model allowing a sensor node to infer the trustworthiness of another node.

Nowadays, many researchers have developed trust models to build up trust relationships among sensor nodes [1]. For example, in a distributed Reputation-based Framework for Sensor Networks (RFSN) is first proposed for WSNs. Two key building blocks of RFSN are Watchdog and Reputation System. Watchdog is responsible for monitoring communication behaviours of neighbour nodes. Reputation System is responsible for maintaining the reputation of a sensor node. The trust value is calculated based on the reputation value. However, in RFSN, only the direct trust is calculated while the recommendation trust is ignored.

A Parameterized and Localized trust management Scheme (PLUS) is proposed in [2]. In PLUS, both personal reference and recommendation are used to build reasonable trust relationship among sensor nodes. Whenever a judge node (the node which performs trust evaluation) receives a packet from suspect node (the node which is in radio range of the judge node and will be evaluated), it always check the integrity of the packet. If the integrity check fails, the trust value of suspect node will be decreased irrespective of whether it was really involved in malicious behaviours or not. Therefore, suspect node may get unfair penalty. Another similar trust evaluation algorithm named as Node Behavioural strategies Banding belief theory of the Trust Evaluation algorithm (NBBTE) is proposed based on behaviour strategy banding D-S belief theory [3].

NBBTE algorithm first establishes various trust factors depending on the communication behaviours between two neighbour nodes. Then, it applies the fuzzy set theory to measure the direct trust values of sensor nodes. Finally, considering the recommendation of neighbour nodes, D-S evidence theory method is adopted to obtain integrated trust value instead of simple weighted-average one. To the best of our knowledge, NBBTE is the first proposed algorithm which establishes various trust factors depending on the communication behaviours to evaluate the trustworthiness of sensor nodes. Therefore, NBBTE is chosen as the comparing algorithm in this paper.

From the literature on this topic, we can find that: 1) In the current research work, the assessment of trust values for sensor nodes is mainly based on the communication (successful and unsuccessful communications) point of view. In fact, just considering the communication behavior, we cannot decide whether a sensor node can be trusted or not. Besides the communication behavior, other trust metrics such as the energy level should also be taken into account to calculate the trustworthiness of sensor nodes. In addition, an efficient trust model should deal with uncertainty caused by noisy communication channels and unstable sensor nodes' behaviors. 2) There are two common ways to establish trust in WSNs: calculating direct trust based on direct interactions and calculating indirect trust value based on recommendation from the third party. However, not all the third parties are trusty and not all the recommendations are reliable. Thus, a discriminate analysis about the third party and recommendation is essential. 3) Most existing studies only provide the trust assessment for neighbor nodes. However, in real applications, a sensor node sometimes needs to obtain the trust value of the non-neighbor nodes. For example, in some routing protocols (e.g., TPGF Plus) or localization algorithms (e.g., improved LMAT algorithm), sensor nodes need the information f the two-hop neighbor nodes to establish the routing or localize themselves. Therefore, providing the trust assessment for non-neighbor nodes becomes very important. 4) Because of the dynamic topology, the trust relationship between sensor nodes constantly changes in WSNs. Trust is a dynamic phenomenon and changes with time and environment conditions. However, most existing trust models do not solve the trust dynamic problem. The evolution of trust over time is another problem that needs further study. In order to solve the above-mentioned problems, we propose an efficient distributed trust model (EDTM). The proposed EDTM can evaluate the trust relationships between sensor nodes more precisely and can prevent security breaches more effectively.

Due to the wireless features of WSNs, it needs a distributed trust model without any central node, where neighbor nodes can monitor each other. In addition, an efficient trust model is required to handle trust related information in a secure and reliable way. In this paper, a distributed and efficient trust model named EDTM was proposed. During the EDTM, the calculation of direct trust, recommendation trust and indirect trust are discussed. Furthermore, the trust propagation and update are studied. Simulation results show that EDTM is an efficient and attack-resistanttrust model. However, how to select the proper value of the weight and the defined threshold is still a challenging problem, that will considered here.

II. RELATED WORKS

A Distributed Three-Hop Routing Protocol Since BSes are connected with a wired backbone,we assume that there are no bandwidth and power constraints on transmissions between BSes. We use intermediate nodes to denote relay nodes that function as gateways connecting an infrastructure wireless network and a mobile ad-hoc network. We assume every mobile node is dual-mode; that is, it has ad-hoc network interface such as a WLAN radio interface and infrastructure network interface such as a 3G cellular interface.

DTR aims to shift the routing burden from the ad-hoc network to the infrastructure network by taking advantage of widespread base stations in a hybrid wireless network. Rather than using one multi-hop path to forward a message to one BS, DTR uses at most two hops to relay the segments of a message to different BSes in a distributed manner, and relies on BSes to combine the segments. Figure demonstrates the process of DTR in a hybrid wireless network. We simplify the routings in the infrastructure network for clarity. As shown in the figure, when a source node wants to transmit a message stream to a destination node, it divides the message stream into a number of partial streams called segments and transmits each segment to a neighbour node.

The majority of outsider attacks against sensor network routing protocols can be prevented by simple link layer encryption and authentication using a globally shared key. Major classes of attacks not countered by link layer encryption and authentication mechanisms are wormhole attacks and HELLO flood attacks because, although an adversary is prevented from joining the network, nothing prevents her from using a wormhole to tunnel packets sent by legitimate nodes in one part of the network to legitimate nodes in another part to convince them they are neighbors or by amplifying an overheard broadcast packet with sufficient power to be received by every node in the network.

Link layer security mechanisms using a globally shared key are completely ineffective in presence of insider attacks or compromised nodes. Insiders can attack the network by spoofing or injecting bogus routing information, creating sinkholes, selectively forwarding packets, using the Sybil attack, and broadcasting HELLO floods. More sophisticated defense mechanisms are needed to provide reasonable protection against wormholes and insider attacks. We focus on countermeasures against these attacks in the remaining sections. Security is a broadly used term encompassing the characteristics of authentication, integrity, privacy, nonrepudiation, and anti-playback [4]. The more the dependency on the information provided by the networks has been increased, the more the risk of secure transmission of information over the networks has increased.

For the secure transmission of various types of information over networks, several cryptographic, steganographic and other techniques are used which are well known. The network security fundamentals and how the techniques are meant for wireless sensor networks.

The encryption-decryption techniques devised for the traditional wired networks are not feasible to be applied directly for the wireless networks and in particular for wireless sensor networks. WSNs consist of tiny sensors which really suffer from the lack of processing, memory and battery power. Applying any encryption scheme requires transmission of extra bits, hence extra processing, memory and battery power which are very important resources for the sensors' longevity.Applying the security mechanisms such as encryption could also increase delay, jitter and packet loss in wireless sensor networks [4].

While cryptography aims at hiding the content of a message, steganography aims at hiding the existence of the message. Steganography is the art of covert communication by embedding a message into the multimedia data (image, sound, video, etc.) [4]. The main objective of steganography is to modify the carrier in a way that is not perceptible and hence, it looks just like ordinary. It hides the existence of the covert channel, and furthermore, in the case that we want to send a secret data without sender information or when we want to distribute secret data publicly, it is very useful. However, securing wireless sensor networks is not directly related to steganography and processing multimedia data (like audio, video) with the inadequate resources [4] of the sensors is difficult and an open research issue.

Physical layer secure access in wireless sensor networks could be provided by using frequency hopping. A dynamic combination of the parameters like hopping set (available frequencies for hopping), dwell time (time interval per hop) and hopping pattern (the sequence in which the frequencies from the available hopping set is used) could be used with a little expense of memory, processing and energy resources. Important points in physical layer secure access are the efficient design so that the hopping sequence is modified in less time than is required to discover it and for employing this both the sender and receiver should maintain a synchronized clock. A scheme as proposed in [4] could also be utilized which introduces secure physical layer access employing the singular vectors with the channel synthesized modulation.

Most of the threats and attacks against security in wireless networks are almost similar to their wired counterparts while some are exacerbated with the inclusion of wireless connectivity. In fact, wireless networks are usually more vulnerable to various security threats as the unguided transmission medium is more susceptible to security attacks than those of the guided transmission medium. The broadcast nature of the wireless communication is a simple candidate for eavesdropping. In most of the cases various security issues and threats related to those we consider for wireless ad hoc networks are also applicable for wireless sensor networks. These issues are well-enumerated in some past researches [5], and also a number of security schemes are already been proposed to fight against them. However, the security mechanisms devised for wireless ad hoc networks could not be applied directly for wireless sensor networks because of the architectural

Attacks against wireless sensor networks could be broadly considered from two different levels of views. One is the attack against the security mechanisms and another is against the basic mechanisms (like routing mechanisms). Here we point out the major attacks in wireless sensor networks.

Denial of Service (DoS) is produced by the unintentional failure of nodes or malicious action. The simplest DoS attack tries to exhaust the resources available to the victim node, by sending extra unnecessary packets and thus prevents legitimate network users from accessing services or resources to which they are entitled. DoS attack is meant not only for the adversary's attempt to subvert, disrupt, or destroy a network, but also for any event that diminishes a network's capability to provide a service. In wireless sensor networks, several types of DoS attacks in different layers might be performed. At physical layer the DoS attacks could be jamming and tampering, at link layer, collision, exhaustion, unfairness, at network layer, neglect and greed, homing, misdirection, black holes and at transport layer this attack could be performed by malicious flooding and desynchronization. The mechanisms to prevent DoS attacks include payment for network resources, pushback, strong authentication and identification of traffic.

In a sensor network, sensors monitor the changes of specific parameters or values and report to the sink according to the requirement. While sending the report, the information in transit may be altered, spoofed, replayed again or vanished. As wireless communication is vulnerable to eavesdropping, any attacker can monitor the traffic flow and get into action to interrupt, intercept, modify or fabricate [5] packets thus, provide wrong information to the base stations or sinks. As sensor nodes typically have short range of transmission and scarce resource, an attacker with high processing power and larger communication range could attack several sensors.

Our second class of sensor network application is security monitoring. Security monitoring networks are composed of nodes that are placed at fixed locations throughout an environment that continually monitor one or more sensors to detect an anomaly. A key difference between security monitoring and environmental monitoring is that security networks are not actually collecting any data. This has a significant impact on the optimal network architecture. Each node has to frequently check the status of its sensors but it only has to transmit a data report when there is a security violation.

In general, complete application scenarios contain aspects of all three categories. For example, in a network designed to track vehicles that pass through it, the network may switch between being an alarm monitoring network and a data collection network. During the long periods of inactivity when no vehicles are present, the network will simply perform an alarm monitoring function. Each node will monitor its sensors waiting to detect a vehicle. Once an alarm event is detected, all or part of the network, will switch into a data collection network and periodically report sensor readings up to a base station that track the vehicles progress. Because of this multi-modal network behavior, it is important to develop a single architecture that and handle all three of these application scenarios.

III. SYSTEM MODELS AND PROBLEM STATEMENT

Security is an important issue for wireless sensor Networks, especially for security sensitive applications. To secure an wireless sensor network, we need to consider the following attributes as criteria to measure security which include availability, confidentiality, integrity, authentication and nonrepudiation. In this paper, we present how to select the proper value of the weight and the defined threshold which we plan to address here.Itis done using RSA algorithm, Trust models have been recently suggested as an effective security mechanism for Wireless Sensor Networks (WSNs). Considerable research has been done on modeling trust. However, most current research works only take communication behavior into account to calculate sensor nodes' trust value, which is not enough for trust evaluation due to the widespread malicious attacks. In this paper, we propose an Efficient Distributed Trust Model (EDTM) for WSNs. First, according to the number of packets received by sensor nodes, direct trust and recommendation trust are selectively calculated. Then, communication trust, energy trust and data trust are considered during the calculation of direct trust. Furthermore, trust reliability and familiarity are defined to improve the accuracy of recommendation trust. The proposed EDTM can evaluate trustworthiness of sensor nodes more precisely and prevent the security breaches more effectively. Simulation results show that EDTM outperforms other similar models, e.g., NBBTE trust model. For providing more security here we use the technique RSA algorithmfor privacy and authentication. Advantages of Proposed System:1)It can prevent security breaches more effectively.2) Provide more security.3) Trusted key exchange.4) Increase the packet delivery ratio.

To efficiently compute the trust values on sensor nodes,we first need a clear understanding of the trust definition and the various trust properties that are adopted in a trust model.

Trust

There are several definitions given to trust in the literature [10]. Trust is always defined by reliability, utility, availability, risk, quality of services and other concepts. Here, trust is defined as a belief level that one sensor node puts on another node for a specific action according to previous observation of behaviours. That is, the trust value is used to reflect whether a sensor node is willing and able to act normally in WSNs. In this paper, a trust value ranges from 0 to 1. A value of 1 means completely trustworthy and 0 means the opposite.

Direct trust

Direct trust is a kind of trust calculated based on the direct communication behaviors. It reflects the trust relationship between two neighbor nodes. Recommendation trust. As mentioned above, since the recommendations from third parties are not always reliable, we need an efficient mechanism to filter the recommendation information. The filtered reliable recommendations are calculated as the recommendation trust.

Indirect trust

When a subject node cannot directly observe an object nodes' communication behaviours, indirect trust can be established. The indirect trust value is gained based on the recommendations from other nodes.Based on[6], we can conclude that there are three main properties of trust: asymmetry, transitivity and composability. Asymmetry implies that if node A trusts node B, it does not necessarily mean that node B trusts node A. Transitivity means the trust value can be passed along a path of trusted nodes. If node A trusts node B and node B trusts node C, it can be inferred that node A trusts node C at a certain level. The transitivity is a very important property in trust calculation between two non-neighbor nodes. Composability implies that trust values received from multiple available paths can be composed together to obtain an integrated value.

The Structure Of EDTM

The overall architecture of EDTM. When we say node B is trustworthy or untrustworthy for node A, there is a trust model between node A and node B. As shown in Fig. 1, EDTM consists of two main components: one-hop trust model and multi-hop trust model which includes the following six components: direct trust module, recommendation trust module, indirect trust module, integrated trust module, trust propagation module and trust update module. When a subject node wants to obtain the trust value of an object, it first checks its recorded list of neighbor nodes. If the ID of the object node is in the list of neighbor nodes, the one-hop trust model is triggered. Otherwise, the multi-hop trust model is started.

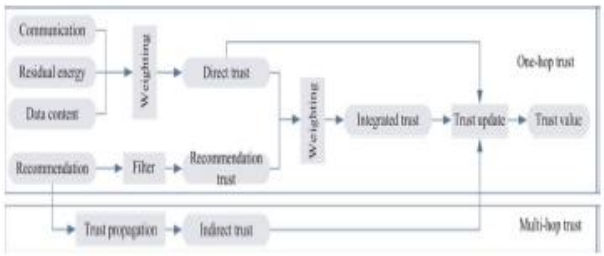


Fig. 1. The EDTM structure

In the one hop trust model, if the trust is calculated based on node B's direct experiences with node A completely,this model is called direct trust model. Otherwise, the recommendation trust module is built. In the multi-hop trust model, once the subject node A receives recommendations from other nodes about the object node B, indirect trust model can be established. In current trust models, the direct trust and recommendation are always used to evaluate the trustworthiness of sensor nodes. The direct trust is directly calculated based on the communication behaviors between two neighbor nodes.

However, due to malicious attacks, using only direct trust to evaluate sensor nodes is not accurate. Thus, the recommendation from other sensor nodes is needed to improve the trust evaluation. In addition, if the number of communication packets between two neighbor nodes is too small, it is difficult to decide whether an object node is good or bad based on only few interactions. Therefore, in the one-hop trust model, we define a threshold of communication packets Thnum. If the communication packets between the subject and object nodes are higher than the threshold Thnum, only the direct trust is calculated. Otherwise, the recommendations from the recommenders are needed for the object's trust evaluation. In the multi-hop trust model, the subject node first needs to select a set of recommenders. Then, the indirect trust is calculated based on recommendations and trust propagation. Next, we describe the detail calculation of direct, recommendation, and indirect trust.

IV. PROPOSED DETECTION SCHEME

4.1 Trust Calculation in EDTM

4.1.1 The Calculation Of Direct Trust

Unlike prior work, we compose our direct trust by considering communication trust, energy trust and data trust. The sensor nodes in WSNs usually collaborate and communicate with neighbor nodes to perform their tasks. Therefore, the communication behaviors are always checked to evaluate whether the sensor node is normal or not. However, due to the nature of wireless communication, there are many reasons resulting in the packets loss and the communications between sensor nodes are unstable. The unsuccessful communication maybe caused by malicious nodes or unstable communication channel. Therefore, just evaluating the communication behaviors is not enough for trust evaluation.

In addition, it is generally known that all communications in WSNs will consume a certain amount of energy to transmit some data packets or any information. If there are malicious nodes in WSNs, the abnormal energy will be consumed or the transmitted data packets will be falsified to conduct malicious attacks. Therefore, communication trust, energy trust and data trust are defined in EDTM. The communication trust reflects if a sensor node can cooperatively execute the intended protocol. The energy trust is used to measure if a sensor node is competent in performing its intended functions or not. The data trust is the trust assessment of the fault tolerance and consistency of data, which affects the trust of the sensor nodes that create and manipulate the data.

4.1.2 Calculation Of The Communication Trust

The information on a sensor node's prior behavior is one of the most important aspects of the communication trust. However, communication channels between two sensor nodes are unstable and noisy, thus monitoring sensor node's behaviors in WSNs based on previous communication behaviors involves considerable uncertainty. To deal with this uncertainty, we adopt a Subjective Logic framework. The trust value in SL framework is a triplet T ={b, d, u } where b, d and u correspond to belief, disbelief and uncertainty respectively, b, d, u ϵ [0, 1] ; b + d + u = 1. Following the trust model based on Subjective Logic framework the communication trust Tcom is calculated based on successful (s) and unsuccessful (f) communication packets:

$$T_{com} = \frac{2b + u}{2}$$

$$\text{where } b = \frac{s}{s+f+1}, u = \frac{1}{s+f+1}$$

4.1.3 Calculation Of The Energy Trust

Energy is an important metric in WSNs since sensor nodes are extremely dependent on the amount of energy they have. Malicious nodes always consume abnormal energy to launch malicious attacks. For example, malicious nodes which conduct DoS attack consume much more energy than normal nodes while selfish nodes consume less energy. Therefore, we use energy as a QoS trust metric to measure if a sensor node is selfish or maliciously exhaust additional energy. Using an energy prediction model, sensor nodes' energy consumption in different periods can be obtained. If the environment conditions do not change much, the energy consumption rate of normal nodes can maintain a stable value.

First, an energy threshold Θ is defined. When the residual energy Eres of one sensor node falls below the threshold value, the sensor node is not competent enough (do not have adequate energy) to perform its intended function. Thus, the energy trust of the sensor node is considered to be 0. Otherwise, The energy trust is calculated based on the energy consumption rate pene, pene ϵ [0, 1]. The higher the energy consumption rate pene is, the less residual energy remains, which ultimately leads to a smaller ability of sensor nodes to complete the task. Thus, the trust values of the sensor nodes are considered to be smaller. The energy trust is calculated by:

$$T_{ene} = \begin{cases} 1 - p_{ene}, & if\ E_{res} \geq \theta, \\ 0, & else, \end{cases}.$$

Where p_{ene} is calculated based on the Ray Projection method. For a object node, if the energy consumption rate in n previous time slots is $P_{ene} = (p_{ene}(1), p_{ene}(2), \ldots, p_{ene}(n))$ and the energy consumption rate in current time slot is pene(n+1), according to the Ray Projection method, the change of energy consumption rate in each time slot is first calculated by $k_i = p_{ene}(i) – p_{ene}(i-1)$, where i = 2, 3, . . ., n. Then, the subject node chooses k_i with the same plus or minus number as k_n and calculate $k_n - k_i$. Place the results of $k_n - k_i$ in an arrangement according to the order from small to large and label as (d_i, l), where $d_i = k_n - k_i$ and l is the labeled position of di in the arrangement. Finally, we obtain $p_{ene}(l) = p_{ene}(n) + k_i + 1$. The minimum value of $p_{ene}(l)$ is chosen as the predicted energy consumption rate $p_{ene}(n+1) = \min(p_{ene}(l))$.

4.1.4 Calculation Of The Data Trust

Following the idea introduced in the trust of the data affects the trust of the network nodes that created and manipulated the data, and vice-versa, we introduce the evaluation of data trust in this section. The data packets have spatial correlation, that is, the packets sent among neighbor nodes are always similar in the same area. The data value of these packets in general follows some certain. An distribution, such as a normal distribution .For the sake of simplicity, in this paper, we also model the distribution of the data as a normal distribution. For a set of data, the probability density function is , where x is the attribute value v_d of a data item, and m and s are mean and variance of the data, respectively. Since the mean m of a set of data is the most representative value that reflects the value similarity of the data, the mean is supposed to have the highest trust value. If the value of a data item is close to the mean, the trust value of this data is relatively high, and vice-versa. Therefore, the trust value of the data item is defined as:

$$T_{data} = 2\left(0.5 - \int_{\mu}^{v_d} f(x)\, dx\right) = 2\int_{v_d}^{\infty} f(x)\, dx.$$

Based on the communication trust T_{com}, the energy trust T_{ene} and the data trust T_{data}, we can obtain the direct trust between two neighbor nodes as:

T_n-direct = $w_{com}T_{com} + w_{ene}T_{ene} + w_{data}T_{data}$;

where wcom, wene and wdata are the weight values of the communication trust, energy trust and data trust respectively, w_{com} ϵ [0, 1] , w_{ene} ϵ [0, 1] , w_{data} ϵ [0, 1] and $w_{com} + w_{ene} + w_{data} = 1$.

4.1.5 Calculation Of The Recommendation Trust

The recommendation trust is a special type of direct trust. When there are no direct communication behaviors between transmit some data packets or any information. If there are malicious nodes in WSNs, the abnormal energy will be consumed or the transmitted data packets will be falsified to conduct malicious attacks.

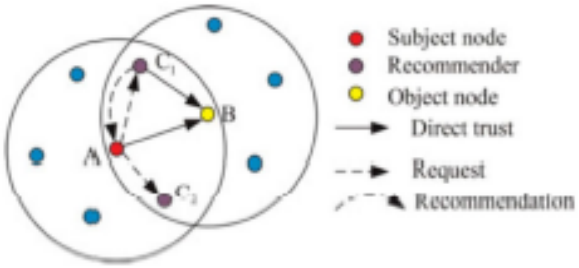


Fig.2 Calculation of the Recommendation Trust

Therefore, communication trust, energy trust and data trust are defined in EDTM. The communication trust reflects if a sensor node can cooperatively execute the intended protocol. The energy trust is used to measure if a sensor node is competent in performing its intended functions or not. The data trust is the trust assessment of the fault tolerance and consistency of data, which affects the trust of the sensor nodes that create and manipulate the data.

As shown in Fig. 2, when a subject node A wants to obtain the recommendations of an object node B. The subject node A first checks its trust records and then selects a set of common neighbor nodes of node A and node B as the recommenders C1;C2; . . . ;Cn, which have the trust value larger than the threshold 0.5. Subsequently, subject node A transmits a recommendation request message to the selected recommenders through multi-casting. Obviously, the identity of node B should be added into the recommendation request. Upon receiving a request message, the qualified nodes will reply if they have recommendation of node B. Based on the recommendations, the subject node A filters the false recommendation and compute the recommendation trust of node B.

4.1.6 Calculation Of The Recommendation Reliability

During the calculation of the recommendation trust, the recommendations from malicious neighbor nodes are first isolated by choosing the trust recommenders. However, not all the recommendations from the recommenders are reliable. Therefore, when the subject node receives several recommendations from neighbor nodes, it will first check whether the recommendations are true or false. This judgment can be done by outlier detection schemes (e.g., checking consistency among multiple recommendations). We consider a simple checking method among multiple recommendations by defining the recommendation reliability T_{rel}. T_{rel} is calculated as follows:

$$T_{rel} = 1 - \left| T_{C_i}^{B} - T_{ave}^{B} \right|,$$

Where T_{Ci}^{B} is the recommendation value of object node B reported by recommender Ci, and TB ave is the average value of all the recommendations.

4.1.7 Calculation Of The Recommendation Familiarity

Generally, the higher trust value of the recommender, the more important recommendation is. Intuitively, it seems to be reasonable. However, it is questionable that nodes with higher trust values give more honest recommendations. Therefore, we introduce the concept of relationship familiarity, which is based on the age of the relationship between two nodes. The concept of familiarity allows sensor nodes to give more importance to recommendations sent by long-term neighbor nodes rather than short term neighbor nodes.

4.1.8 Calculation Of The Indirect Trust

WSNs are multi-hop networks, when there are no direct communications between subject and object nodes, indirect trust can be established since trust is transitive. In this paper, the calculation of indirect trust includes two steps: 1) the first step is to find multi-hop recommenders between subject and object nodes, and 2) the second step is the trust propagation which aims at computing the direct trust.

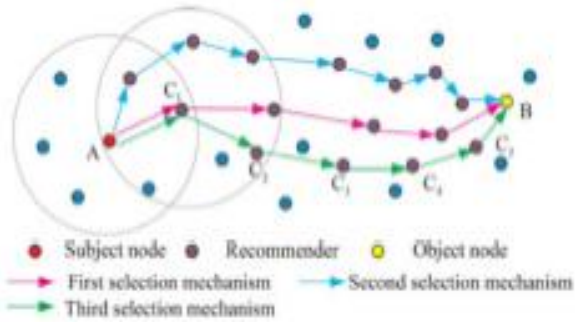


Fig.3 Calculation of the Indirect Trust

The path from the subject node to the object node established by the recommenders is named as Trust Chain. As shown in Fig.3.3 , based on the location information of sensor nodes, we observe three different kinds of mechanisms for choosing the recommender in this paper: 1) finding a recommender which is closest to the object node to save energy consumption, 2) finding a recommender which has the highest trust value to guarantee the reliability of Trust Chain and 3) finding an optimal Trust Chain by both considering the distance information and the trust value. The first selection mechanism can find the shortest Trust Chain, thus the communication overhead for indirect trust calculation can be minimized. However, in this case, the indirect trust evaluation

is not accurate because malicious nodes maybe chosen as recommenders. While the second selection mechanism can choose the most believable Trust Chain but this Trust Chain is not energy efficient. Relatively speaking, the third selection mechanism is the best one.

Link error and malicious packet dropping are two sources for packet losses in multi-hop wireless ad hoc network.While observing a sequence of packet losses in the network, we are interested in determining whether the losses are caused by link errors only, or by the combined effect of link errors and malicious drop. We are especially interested in the insider-attack case, whereby malicious nodes that are part of the route exploit their knowledge of the communication context to selectively drop a small amount of packets critical to the network performance. Because the packet dropping rate in this case is comparable to the channel error rate, conventional algorithms that are based on detecting the packet loss rate cannot achieve satisfactory detection accuracy. To improve the detection accuracy, we propose to exploit the correlations between lost packets. Furthermore, to ensure truthful calculation of these correlations, we develop a homomorphic linear authenticator (HLA) based public auditing architecture that allows the detector to verify the truthfulness of the packet loss information reported by nodes. This construction is privacy preserving, collusion proof, and incurs low communication and storage overheads. To reduce the computation overhead of the baseline scheme, a packet-block-based mechanism is also proposed, which allows one to trade detection accuracy for lower computation complexity. Through extensive simulations, we verify that the proposed mechanisms achieve significantly better detection accuracy than conventional methods such as a maximum-likelihood based detection.

4.2 Packet Loss And Link Control

The proposed mechanism is based on detecting the correlations between the lost packets over each hop of the path.The basic idea is to model the packet loss process of a hop as a random process alternating between 0 (loss) and 1 (no loss). Specifically, consider that a sequence of M packets that are transmitted consecutively over a wireless channel. By observing whether the transmissions are successful or not, the receiver of the hop obtains a bitmap ($a_1; \ldots; a_M$), where aj €{0, 1} for packets j = 1, . . .,M. The correlation of the lost packet is calculated as the auto-correlation function of this bitmap. Under different packet dropping conditions, i.e., link-error versus malicious dropping, the instantiations of the packet-loss random process should present distinct dropping patterns (represented by the correlation of the instance). This is true even when the packet loss rate is similar in each instantiation. To verify this property, in Figure we have simulated the auto-correlation functions of two packet loss processes, one caused by 10 percent link errors, and the other by 10 percent link errors plus 10 percent malicious uniformly-random packet dropping.

4.2.1 Setup Phase

This phase takes place right after route PSD is established, but before any data packets are transmitted over the route. In this phase, S decides on a symmetric-key crypto-system ($encrypt_{key}$; $decrypt_{key}$) and K symmetric keys $key_1; \ldots; key_K$, where encryptkey and decryptkey are the keyed encryption and decryption functions, respectively. S securely distributes decryptkey and a symmetric key keyj to node nj on PSD, for j = 1; . . .;K. Key distribution may be based on the public-key crypto-system such as RSA: S encrypts key_j using the public key of node nj and sends the cipher text to nj. nj decrypts the cipher text using its private key to obtain keyj. S also announces two hash functions, H1 and HMAC key , to all nodes in PSD. H1 is unkeyed while HMAC key is a keyed hash function that will be used for message authentication purposes later on.

Besides symmetric key distribution, S also needs to set up its HLA keys. Let e : G x G ->GT be a computable bilinear map with multiplicative cyclic group G and support Zp, where p is the prime order of G, i.e., for all a , b € G and q1, q2 € Zp, $e(a^{q1}; b^{q2}) = e(a; b)^{q1q2}$. Let g be a generator of G. H2(.) is a secure map-to-point hash function: which maps strings uniformly to G. S chooses a random number x € Zp and computes v = gx. Let u be another generator of G. The secret HLA key is sk = x and the public HLA key is a tuple pk = (v; g; u).

4.2.2 Packet Transmission Phase

After completing the setup phase, S enters the packet transmission phase. S transmits packets to PSD according to the following steps. Before sending out a packet Pi, where i is a sequence number that uniquely identifies Pi, S computes ri = H1(Pi) and generates the HLA signatures of ri for node nj, as follows:

$$s_{ji} = \left[H_2(i||j)u^{r_i}\right]^x, \quad \text{for } j = 1, \ldots, K,$$

where || denotes concatenation. These signatures are then sent together with Pi to the route by using a one-way chained encryption that prevents an upstream node from deciphering the signatures intended for downstream nodes. More specifically, after getting sji for j = 1; . . .;K, S iteratively computes the following:

$$\begin{aligned}
\tilde{s}_{Ki} &= encrypt_{key_K}(s_{Ki}),\\
\tau_{Ki} &= \tilde{s}_{Ki}||MAC_{key_K}(\tilde{s}_{Ki}),\\
\tilde{s}_{K-1i} &= encrypt_{key_{K-1}}(s_{K-1i}||\tau_{Ki}),\\
\tau_{K-1i} &= \tilde{s}_{K-1i}||MAC_{key_{K-1}}(\tilde{s}_{K-1i}),\\
&\vdots\\
\tilde{s}_{ji} &= encrypt_{key_j}(s_{ji}||\tau_{j+1i}),\\
\tau_{ji} &= \tilde{s}_{ji}||MAC_{key_j}(\tilde{s}_{ji}),\\
&\vdots\\
\tilde{s}_{1i} &= encrypt_{key_1}(s_{1i}||\tau_{2i}),\\
\tau_{1i} &= \tilde{s}_{1i}||MAC_{key_1}(\tilde{s}_{1i}),
\end{aligned}$$

where the message authentication code (MAC) in each stage j is computed according to the hash function HMAC keyj. After getting t1i, S puts Pijjt1i into one packet and sends it to node n1. When node n1 receives the packet from S, it extracts Pi, ~s1i, and MACkey1 (~s1i) from the received packet. Then, n1 verifies the integrity of ~s1i by testing the following equality:

$$MAC_{key_1}(\tilde{s}_{1i}) = H^{MAC}_{key_1}(\tilde{s}_{1i}).$$

If the test is true, then n1 decrypts ~s1i as follows:

$$decrypt_{key1}\ \eth(s1i) = s1i\|t2i$$

Then, n1 extracts s1i and t2i from the decrypted text. It stores ri = H1(Pi) and s1i in its proof-of-reception database for future use. This database is maintained at every node on PSD. It can be considered as a FIFO queue of size M, which records the reception status for the most recent M packets sent by S. Finally, n1 assembles Pijjt2i into one packet and relays this packet to node n2. In case the test in (5) fails, n1 marks the loss of Pi in its proof-of-reception database and does not relay the packet to n2.

The above process is repeated at every intermediate node nj, j = 1; . . .;K. As a result, node nj obtains ri and its HLA signature sji for every packet Pi that the node has received, and it relays Pi‖ tj+1i to the next hop on the route. The last hop, i.e., node nK, only forwards Pi to the destination D. As proved in Theorem 4 in Section 4.3, the special structure of the one-way chained encryption construction in (4) dictates that an upstream node on the route cannot get a copy of the HLA signature intended for a downstream node, and thus the construction is resilient to the collusion model defined in Section 3.2. Note that here we consider the verification of the integrity of Pi as an orthogonal problem to that of verifying the tag tji. If the verification of Pi fails, node n1 should also stop forwarding the packet and should mark it accordingly in its proof-of-reception database.

4.2.3 Audit Phase

This phase is triggered when the public auditor Ad receives an ADR message from S. The ADR message includes the id of the nodes on PSD, ordered in the downstream direction, i.e., n1; . . . ; nK, S's HLA public key information pk =(v; g; u), the sequence numbers of the most recent M packets sent by S, and the sequence numbers of the subset of these M packets that were received by D. Recall that we assume the information sent by S and D is truthful, because detecting attacks is in their interest. Ad conducts the auditing process as follows.

A_d submits a random challenge vector ~cj =(cj1; . . . ; cjM) to node nj, j = 1; . . .;K, where the elements cji's are randomly chosen from Zp. Without loss of generality, let the sequence number of the packets recorded in the current proof-of-reception database be P1; . . . ; PM, with PM being the most recent packet sent by S. Based on the information in this database, node nj generates a packet-reception bitmap~bj=(bj1; . . . ; bjM), where bji = 1 if Pi has been received by nj, and bji ¼ 0 otherwise. Node nj then calculates the linear combination.

If the equality holds, then Ad accepts that node nj received the packets as reflected in ~bj. Otherwise, Ad rejects ~bj and judges that not all packets claimed in~bj are actually received by nj, so nj is a malicious node. We prove the correctness of this auditing algorithm. Note that the above mechanism only guarantees that a node cannot understate its packet loss, i.e., it cannot claim the reception of a packet that it actually did not receive. This mechanism cannot prevent a node from overly stating its packet loss by claiming that it did not receive a packet that it actually received. This latter case is prevented by
another mechanism discussed in the detection phase.

4.2.4 Detection Phase

The public auditor Ad enters the detection phase after receiving and auditing the reply to its challenge from all nodes on PSD. The main tasks of Ad in this phase include the following: detecting any overstatement of packet loss at each node, constructing a packet-loss bitmap for each hop, calculating the autocorrelation function for the packet loss on each hop, and deciding whether malicious behavior is present. More specifically, Ad performs these tasks as follows. Given

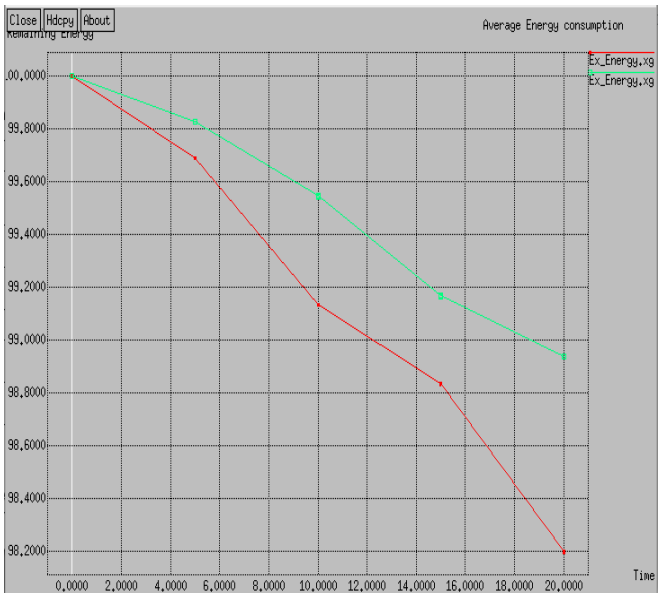
the packet-reception bitmap at each node, ~b1; . . . ; ~bK, Ad first checks the consistency of the bitmaps for any possible overstatement of packet losses. Clearly, if there is no overstatement of packet loss, then the set of packets received at node j þ 1 should be a subset of the packets received at node j, for j = 1; . . .;K _ 1. Because a normal node always truthfully reports its packet reception, the packet-reception bitmap of a malicious node that overstates its packet loss must contradict with the bitmap of a normal downstream node. Note that there is always at least one normal downstream node, i.e., the destination D. So Ad only needs to sequentially scan ~bj's and the report from D to identify nodes that are overstating their packet losses. After checking for the consistency of ~bj's, Ad starts constructing the per-hop packet-loss bitmap ~mj from ~bj_1 and ~b j. This is done sequentially, starting from the first hop from S. In each step, only packets that are lost in the current hop will be accounted for in mj. The packets that were not received by the upstream node will be marked as "not lost" for the underlying hop. Denoting the "lost" packet by 0 and "not lost" by 1, ~mj can be easily constructed by conducting a bit-wise complement-XOR operation of ~bj_1 and ~bj. For example, consider the following simple case with three intermediate nodes (four hops) on the route and with M =10. Suppose that ~b1 =(0; 1; 1; 1; 1; 1; 1; 1; 0; 1), ~b2 = (0; 1; 1; 1; 1; 1; 1; 1; 0; 1), ~b3 = (0; 1; 0; 1; 1; 0; 1; 1; 0; 1), and the destination D reports that~bD = (0; 1; 0; 1; 1; 0; 1; 1; 0; 1). Then the per-hop packet-loss bitmaps are given by ~m1 = (0; 1; 1; 1; 1; 1; 1; 1; 0; 1), ~m2 =(1; 1; 1; 1; 1; 1; 1; 1; 1; 1), ~m3 = (1; 1; 0; 1; 1; 0; 1; 1; 1; 1), and ~m4 =(1; 1; 1; 1; 1; 1; 1; 1; 1; 1).

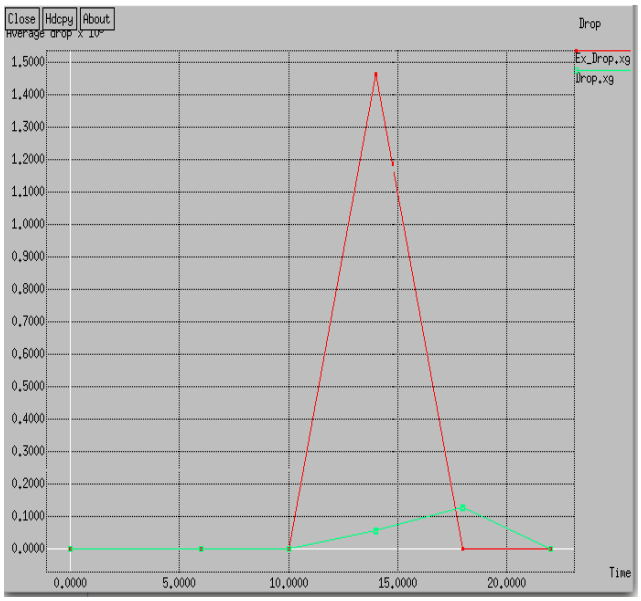
The relative difference €j is then used as the decision statistic to decide whether or not the packet loss over the jth hop is caused by malicious drops. In particular, if €j>=€th, where €th is an error threshold, then Ad decides that there is malicious packet drop over the hop. In this case, both ends of the hop will be considered as suspects, i.e., either the transmitter did not send out the packet or the receiver chose to ignore the received packet. S may choose to exclude both nodes from future packet transmissions, or alternatively, apply a more extensive investigation to refine its detection.

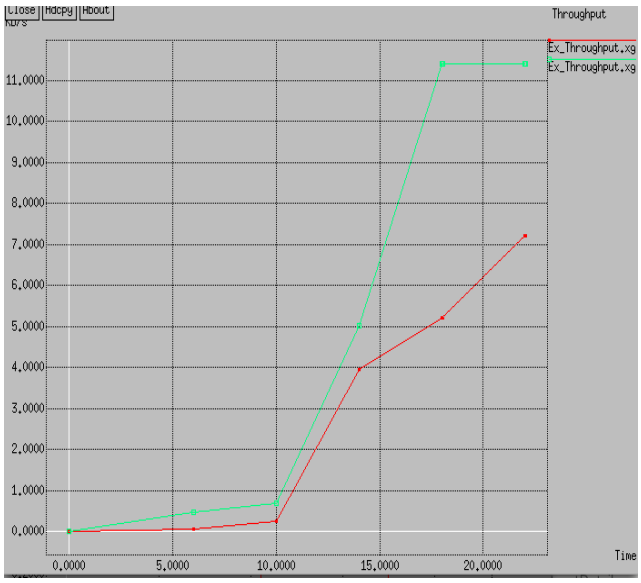
For example, this can be done by combining the neighbor overhearing techniques [12] used in the reputation system. By fusing the testimony from the neighbors of these two nodes, Ad can pin-point the specific node that dropped the packet. Once being detected, the malicious node will be marked and excluded from the route to mitigate its damage. The above detection process applies to one end-to-end path. The detection for multiple paths can be performed as multiple independent detections, one for each path. Although the optimal error threshold that minimizes the detection error is still an open problem, our simulations show that through trial-and-error, one can easily find a good €th that provides a better detection accuracy than the optimal detection scheme

V. PERFORMANCE EVALUATION

In this section, we compare the detection accuracy achieved by the proposed algorithm with the optimal maximum likelihood algorithm, which only utilizes the distribution of the number of lost packets. For given packet-loss bitmaps, the detection on different hops is conducted separately. So, we only need to simulate the detection of one hop to evaluate the performance of a given algorithm. We assume packets are transmitted continuously over this hop, i.e., a saturated traffic environment. We assume channel fluctuations for this hop follow the Gilbert-Elliot model, with the transition probabilities from good to bad and from bad to good given by PGB and PBG, respectively. We consider two types of malicious packet dropping: random dropping and selective dropping.







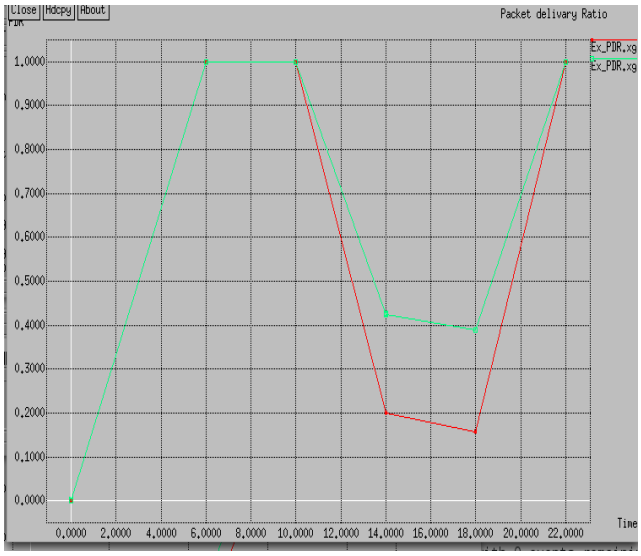


Fig 4 Average Energy Consumption ,Drop,Throughput and Packet Delivery Ratio.

VI. CONCLUSION AND FUTURE SCOPE

The trust model has become important for malicious nodes detection in WSNs. It can assist in many applications such as secure routing, secure data aggregation, and trusted key exchange. Due to the wireless features of WSNs, it needs a distributed trust model without any central node, where neighbor nodes can monitor each other. In addition, an efficient trust model is required to handle trust related information in a secure and reliable way. In this paper, a distributed and efficient trust model named EDTM was proposed. During the EDTM, the calculation of direct trust,recommendation trust and indirect trust are discussed. Furthermore, the trust propagation and update are studied. Simulation results show that EDTM is an efficient and attack-resistant trust model. The proposed EDTM can evaluate trustworthiness of sensor nodes more precisely and prevent the security breaches more effectively. Simulation results show that EDTM outperforms other similar models, e.g., NBBTE trust model. For providing more security here we use the technique RSA algorithmfor privacy and authentication.The main 3 factors used in the communication than the trust model are Link quality,Distance between the nodes and the Security Measures.

However, how to select the proper value of the weight and the defined threshold is still a challenging problem, which we plan to address in our future research endeavors.

REFERENCES

[1] H. Chan and A Perrig, "Protocols in wireless sensor networks," *Comput*., vol. 36, no. 10, pp. 103–105, Oct. 2003.

[2] Y. M. Huang, M. Y. Hsieh, H. C. Chao, S. H. Hung, and J. H. Park, "Distributed three-hop Routing protocols," IEEE J. *Sel. Areas Commun*., vol. 24, no. 7, pp. 400–411, May 2009.

[3] V. C. Gungor, L. Bin, and G. P. Hancke, "Wireless Network Security Analysis CounterMeasures" IEEE *Trans. Ind.Electron*., vol. 57, no. 10, pp. 3557–3564, Oct. 2010.

[4] G. Han, J. Jiang, L. Shu, J. Niu, and H. C. Chao, "Security in Wireless Sensor Network:Issues and Challenges," J. *Comput. Syst. Sci*., vol. 80, no. 3, pp. 602–617, 2014.

[5] S. Ganeriwal, L. K. Balzano, and M. B. Srivastava, "Security Threats and Issues in Wireless Sensor Networks" in Proc. *2nd ACM Workshop Security Ad Hoc Sensor Netw*., 2004, pp. 66–77.

[6] Z. Yao, D. Kim, and Y. Doh, "System architecture of Wireless Sensor Networks ," in Proc. IEEE *Int. Conf. Mobile Adhoc Sensor Syst*., 2008, pp. 437–446.

[7] M. Rabbat and R. Nowak, "Energy Efficient Schemes for Wireless sensor Networks," in Proc. *3rd Int. Symp. Inf. Process. Sens. Netw*., 2004, pp. 20–27.

[8] K. Shao, F. Luo, N. Mei, and Z. Liu, "Routing Techniques in Wireless Sensor Networks" J. *Softw*., vol. 23, no. 12, pp. 3130–3148, 2012.

[9] G. Han, X. Xu, J. Jiang, L. Shu, and N. Chilamkurti, "Trust Based Approach in WSN using an Agent to each Cluster" *KSII Trans. Internet Inf. Syst*., vol. 6, pp. 2992–3007, 2012.

[10] K. Govindan and P. Mohapatra, "Security Solutions for Wireless Sensor Networks," *IEEE Commun. Surveys Tuts*., vol. 14, no. 2, pp. 279–298, 2nd Quarter 2012.