International Journal of Current Advanced Research
ISSN: O: 2319-6475, ISSN: P: 2319-6505, Impact Factor: SJIF: 5.995
Available Online at www.journalijcar.org
Volume 7; Issue 2(J); February 2018; Page No. 10309-10314
DOI: http://dx.doi.org/10.24327/ijcar.2018.10314.1740

Research Article

ENERGY EFFICIENT AND SECURE ROUTING TECHNIQUE FOR WIRELESS SENSOR NETWORKS

Girish.R.Deshpande[1]., Abdull.Rahaiman[2] and Niranjan S.J[3]

[1]Department of CSE, GIT, Belgaum-590008
[2]Department of ECE, PACE, Manglore-572202
[3]Department of CSE, GIT, Tiptur-572202

A R T I C L E I N F O

Article History:
Received 20th November, 2017
Received in revised form 25th December, 2017
Accepted 13th January, 2018
Published online 28th February, 2018

Key words:
Wireless sensor networks, trust management, Security, Implementation cost

A B S T R A C T

The multihop routing in wireless sensor networks (WSNs) offers little protection against identity deception through replaying routing information. An adversary can exploit this defect to launch various harmful or even devastating attacks against the routing protocols, including sinkhole attacks, wormhole attacks, and Sybil attacks. The situation is further aggravated by mobile and harsh network conditions. In this paper, we present a trust-aware, location-based routing protocol which protects the WSN against routing attacks, and also supports large scale WSNs Deployments. The proposed solution has been shown to efficiently detect and avoid malicious nodes and has been implemented in state-of-the-art sensor nodes for a real-life test-bed. As it will be discussed, the limited memory, computational power, energy resources and radio bandwidth of sensor nodes deeply impact the implementation strategy, while additionally, the realities of radio propagation, such as lossy and asymmetric links, require careful evaluation of the routing selection metrics.

INTRODUCTION

Wireless Sensor Networks (WSN) offer efficient, low-cost solutions for a great variety of application domains including military fields, healthcare, homeland security, industry control, intelligent green aircrafts and traffic control in smart roads . Although networking and security technologies are in a mature stage, the limited sensor node resources in terms of memory space, processing power and energy availability, constrain the complexity of the security mechanisms that can be implemented, dictating the need for new protocol approaches design. Due to their distributed nature, WSNs are vulnerable to various attacks [2], including attacks targeting on the disruption of the routing procedure which is accomplished in a cooperative, multi-hop fashion. While the traditional (or the so called "hard") security measures (e.g. encryption, authentication) are quite efficient in mitigating some types of attacks, there are some specific types of attacks that can be better handled by using a reputation and trust-based management scheme (as an example, selfish behavior of a node). In other words, security and trust are tightly coupled and cannot be separated information is derived to guide routing decisions. TARF secures the multi-hop routing in WSNs against intruders exploiting the replay of routing information by evaluating the trust worthiness of neighboring nodes. It identifies such intruders that misdirect noticeable network traffic by their low trustworthiness and routes data through paths circumventing those intruders to achieve satisfactory throughput.

Corresponding author:* **Girish.R.Deshpande
Department of CSE, GIT, Belgaum-590008

Wireless sensor networks (WSNs) [2] are ideal candidates for applications to report detected events of interest, such as military surveillance and forest fire monitoring. A WSN comprises battery-powered senor nodes with extremely limited processing capabilities. With a narrow radio communication range, a sensor node wirelessly sends messages to a base station via a multi-hop path. However, the multi-hop routing of WSNs often becomes the target of malicious attacks. An attacker may tamper nodes physically, create traffic collision with seemingly valid transmission, drop or misdirect messages in routes, or jam the communication channel by creating radio interference [3]. This project focuses on the kind of attacks in which adversaries misdirect network traffic by identity deception through replaying routing information. Based on identity deception, the adversary is capable of launching harmful and hard-to-detect attacks against routing, such as selective forwarding, wormhole attacks, sinkhole attacks and Sybil attacks [4].

LITERATURE REVIEW

Theodore Zahariadis *et.al*., proposed that Ad-hoc personal area networks (PAN) and wireless sensor networks impose

new challenges on the design of security tools which are more imperative than ever due to their unattended operation in open environments. To defend against routing attacks, the implementation of a trust management system is suggested. they presented a trust-aware routing protocol that can efficiently detect and avoid nodes issuing routing attacks based on a distributed trust management system.

Jaydip Sen shown that public key operations are still very expensive to realize in sensor nodes. A public key cryptography can greatly ease the design of security in WSNs, improving the efficiency of private key operations on sensor nodes is highly desirable.

Secure routing protocols for mobile sensor networks: mobility of sensor nodes has a great influence on sensor network topology and thus on the routing protocols. Mobility can be at the base station, sensor nodes, or both. Current protocols assume the sensor network is stationary. New secure routing protocols for mobile sensor networks need to be developed Theodore Zahariadis explained that Wireless sensor networks impose new challenges on the design of security tools. We successfully implemented a trust- aware routing protocol and validated it in real-life sensor nodes proving its implementation feasibility. The conducted tests showed that, although in the literature trust management has been dealt with as an independent functional block and augmenting the trust metrics list has been pursued towards improving security, in real world deployments the added value of implementing a trust management system depends on the protocols employed in the WSN and the way they use the trust information.

NTRU (Nth-degree truncated polynomial ring unit algorithm)

The NTRU Encrypt public key cryptosystem, also known as the NTRU encryption algorithm, is a lattice-based alternative to RSA and ECC and is based on the shortest vector problem in a lattice (i.e. is not known to be breakable using quantum computers). Operations are based on objects in a truncated polynomial ring with convolution multiplication and all polynomials in the ring have integer coefficients and degree at most N-1.

Steps involved in NTRU Key Generation

1. Creating the inverse polynomial of the secret key modulo q, Fq.
2. Creating the inverse polynomial of the secret key modulo p, Fp.
3. Creating the public key, h=p*((Fq)*g) modq.

NTRU Encryption

User A has a message to transmit to B , So A first puts the message in the form of a polynomial m whose coefficients is chosen modulo p say between as –p/2 and p/2. Next A randomly chooses another small polynomial r. This is the binding value which is used to obscure the message. A uses the message m, randomly chosen polynomial r, and B's public key h to compute the polynomial e = r*h + m (modulo q). The polynomial e is the encrypted message which A sends to B.

1. Performing the polynomial multiplication of h*r.
2. *Adding the message m and the modulo reduction is performed by extracting the lower w bits*

The decryption procedure is executed by the following three steps:

1. Performing the polynomial multiplication of a = f * e mod q.
2. Shifting the coefficients of a into the range (-q/2; q/2).
3. Performing the polynomial multiplication d=a*Fp modp.

Software requirements

Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or prerequisites are generally not included in the software installation package and need to be installed separately before the software is installed.

- Java1.4 or higher
- Java Swing – front end
- JDBC –Database connectivity
- UDP-User Datagram Protocol
- TCP-Transmission Control Protocol
- Networking-Socket programming
- ORACLE –Back end
- Windows XP or higher-Operating System

System design

System design is the process of defining the architecture, components, modules, interfaces and data for a system to satisfy specified requirements. One could see it as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis, systems architecture and systems engineering. If the broader topic of product development "blends the perspective of marketing, design, and manufacturing into a single approach to product development," then design is the act of taking the marketing information and creating the design of the product to be manufactured. Systems design is therefore the process of defining and developing systems to satisfy specified requirements of the user.

UML Diagrams

Unified Modeling Language (UML) is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created, by the Object Management Group.

Use Case Diagrams

A use case diagram at its simplest is a graphical representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system.

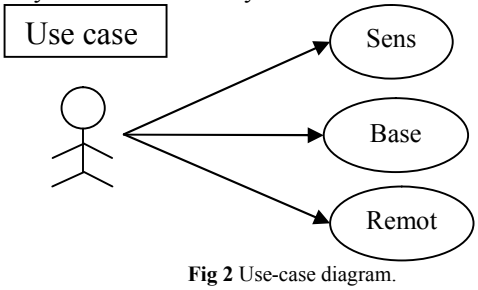

Fig 2 Use-case diagram.

Data Flow Diagram

- A data flow diagram is a graphical representation of the "flow" of data through an information system, modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).
- The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out on these data, and the output data is generated by the system.

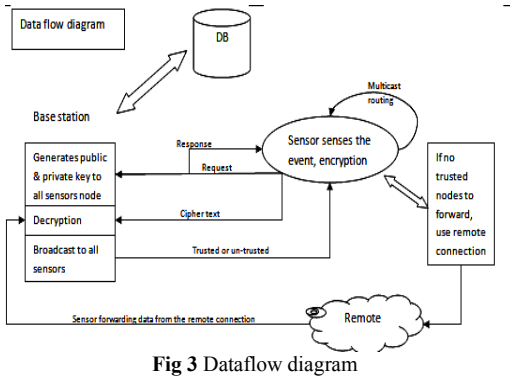

Fig 3 Dataflow diagram

Sequence Diagram

A sequence diagram in a UML is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams typically are associated with use case realizations in the Logical View of the system under development.

Flowcharts

A flow chart is a graphical or symbolic representation of a process. Each step in the process is represented by a different symbol and contains a short description of the process step. The flow chart symbols are linked together with arrows showing the process flow direction.

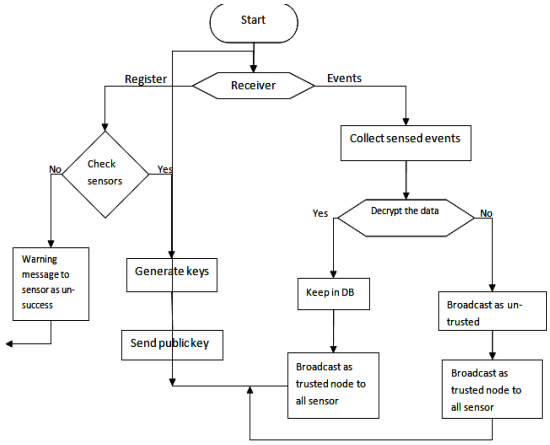

Fig 4 Flow diagram

Implementation

Network Model

In this work, a wireless sensor network composed of a network controller which is a trusted party, a large number of sensor nodes, and many users are considered. It denotes the network controller with the symbol T. Symbol U and N are used to represent the universe of the users and the sensor nodes respectively. Both users and sensor nodes have their unique IDs. Symbol U_i will be used to denote user i, and Ni is defined similarly. The trusted party T can be online or off-line. It comes online merely on necessity basis, e.g., in the case of intruders detected. Each sensor could be a high-end sensor node such as iMote2 which has greater processing capability and a larger memory than conventional sensor nodes. Sensor data could be stored locally or at some designated in-network location using data storage schemes such as TTDD. As is conventionally assumed, in this project consider a user U to have sufficient computational resources to execute some expensive cryptographic operations. In addition, assume there is loose time synchronization among the sensor nodes.

Adversary Model

It considers attackers whose main goal is to obtain sensor data which they are not authorized to access. The adversaries could be either external intruders or network users who are unauthorized to access the target type of data. Due to lack of physical protection, sensor nodes are usually vulnerable to strong attacks. In particular, it considers the adversary with both passive and active capabilities, which can

- Eaves drop all the communication traffics in the WSN.
- Compromise and control a small number of sensor nodes.
- Unauthorized users may collude to compromise the encrypted data.

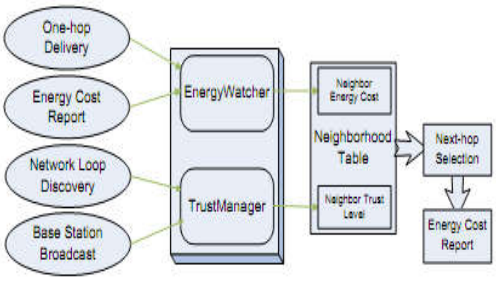

Fig 1 Energy Watcher and Trust Manager

Fig. each node selects a next-hop node based on its neighborhood table, and broadcast its energy cost within its neighborhood. To maintain this neighborhood table, Energy Watcher and Trust Manager on the node keep track of related events (on the left) to record the energy cost and the trust level values of its neighbors.

Routing Procedure

TARF, as with many other routing protocols, runs as a periodic service. The length of that period determines how frequently routing information is exchanged and updated. At the beginning of each period, the base station broadcasts a message about data delivery during last period to the whole network consisting of a few contiguous packets (one packet may not hold all the information). Each such packet has fields

to indicate how many packets are remaining to complete the broadcast of the current message. The completion of the base station broadcast triggers the exchange of energy report in this new period. Whenever a node receives such a broadcast message from the base station, it knows that the most recent period has ended and a new period has just started. No tight time synchronization is required for a node to keep track of the beginning or ending of a period. During each period, the Energy Watcher on a node monitors energy consumption of one-hop transmission to its neighbors and processes energy cost reports from those neighbors to maintain energy cost entries in its neighborhood table; its Trust Manager also keeps track of network loops and processes broadcast messages from the base station about data delivery to maintain trust level entries in its neighborhood table.

Energy Watcher

In this section explains about how a node N's Energy Watcher computes the energy cost ENb for its neighbor b in N's neighborhood table and how N decides its own energy cost ENb

Trust Manager

A node N's Trust Manager decides the trust level of each neighbor based on the following events: discovery of network loops, and broadcast from the base station about data delivery. For each neighbor b of N, T denotes the trust level of b in N's neighborhood table. At the beginning, each neighbor is given a neutral trust level 0.5. After any of those events occurs, the relevant neighbors' trust levels are updated.

Analysis on Energy Watcher and Trust Manager

Now that a node N relies on its Energy Watcher and Trust Manager to select an optimal neighbor as its next hop node, to clarify a few important points on the design of Energy Watcher and Trust Manager. First, the energy cost report is the only information that a node is to passively receive and take as "fact". It appears that such acceptance of energy cost report could be a pitfall when an attacker or a compromised node forges false report of its energy cost. Note that the main interest of an attacker is to prevent data delivery rather than to trick a data packet into a less efficient route, considering the effort it takes to launch an attack. As far as an attack aiming at preventing data delivery is concerned, TARF well mitigates the effect of this pitfall through the operation of Trust manager. Note that the Trust Manager on one node does not take any recommendation from the Trust Manager on another node.

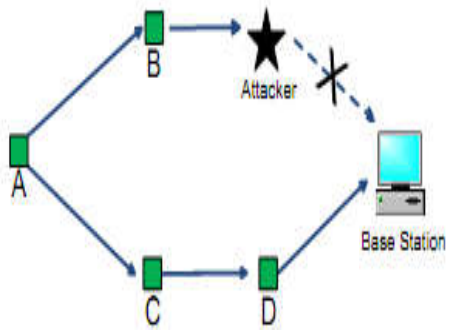

Fig 2 An example to illustrate How Trust Manager works.

If an attacker forges false energy report to form a false route, such intention will be defeated by Trust Manager: when the Trust Manager on one node finds out the many delivery failures from the broadcast messages of the base station, it degrades the trust level of its current next-hop node; when that trust level goes below certain threshold, it causes the node to switch to a more promising next hop node. Second, Trust Manager identifies the low trustworthiness of various attackers misdirecting the multi-hop routing, especially those exploiting the replay of routing information. It is noteworthy that Trust Manager does not distinguish whether an error or an attack occurs to the next-hop node or other succeeding nodes in the route.

Snapshots and Results of the Project

Base station module

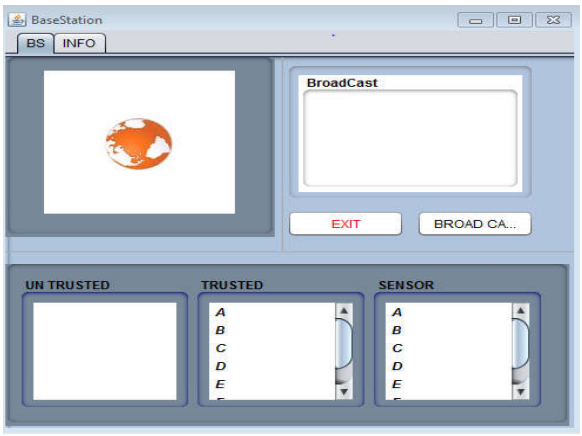

Fig 6.1 base station module

In the above fig 6.1 shows a base station module. This module contains all available trusted sensor nodes in the network. After any of events occurs it will broadcast trusted and untrusted nodes to all sensors which are present in the network.

Sensor node

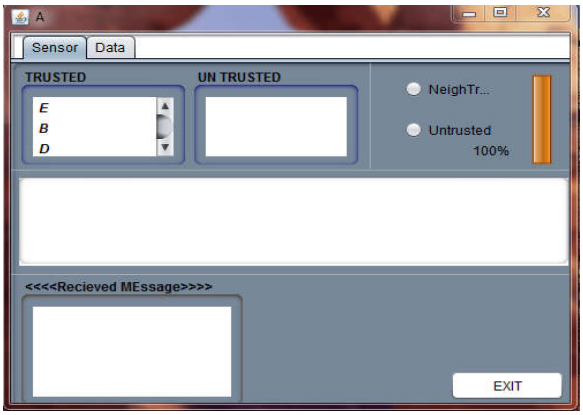

Fig 6.2 sensor module

In above fig 6.2 which shows sensor module 'A', Sensor 'A' contains all its trusted neighbor nodes in the network. In above fig sensors 'E', 'B', 'D's are trusted neighbor nodes to sensor node A. After sending a data the battery level updated every time in the sensor node A.

Sending a message from sensor node A to sensor node E.

Transmission of Data

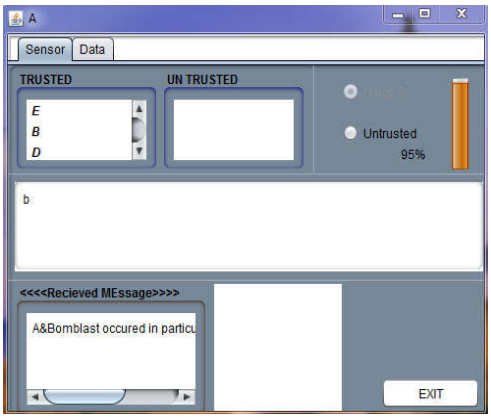

Fig 6.3 sending information

Now sensor node 'A' wants to send a data to a base station through intermediate node E. The node 'A' chooses 'E' as its trusted neighbor node.

The details of sending operations are:
Source node: Sensor node 'A'
Intermediate node: Trusted sensor node 'E'
Destination node: Base station node (BS)

Message details are: "Bomb blast occurred in a particular sensed region at a time 3:00 pm."

After pressing key on key board the sensor node 'A' sends a data to the BS. The available battery power is 95% in sensor node A.

Information stored in sensor node A

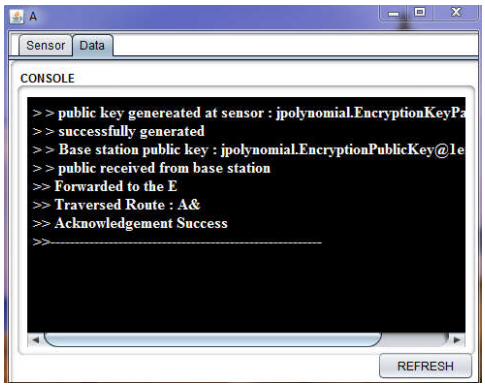

Fig 6.4 information stored in sensor node

The public key generated at a sensor node A is @df574d, the length of the key is 256 byte. The base station public key is generated as @lc8cc59; the public key is received from base station. The intermediate node E receives a data from node A .Node E sends acknowledgement to sensor node A.

Information stored in sensor node E

The node E receives a data from sensor node 'A'. Node 'E' computes its trusted node for successful routing.

The public key generated at a sensor node 'E' is @1d5e94f, the length of the key is 256 byte. The base station public key is generated as @le6f1f4; the public key is received from base station. The intermediate node E sends a data to a base station successfully. Base station returns acknowledgement to node E. Traversed route is A&E&BS.

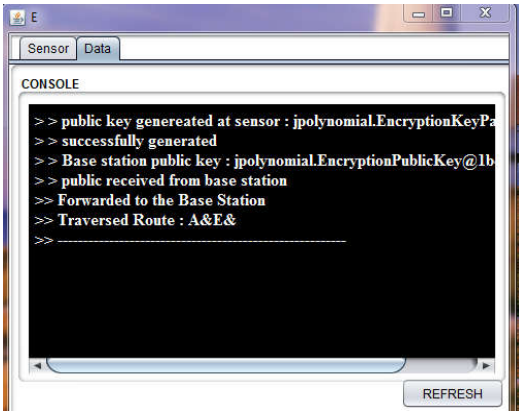

Fig 6.5 information stored in sensor node E

Information stored in Base station (BS)

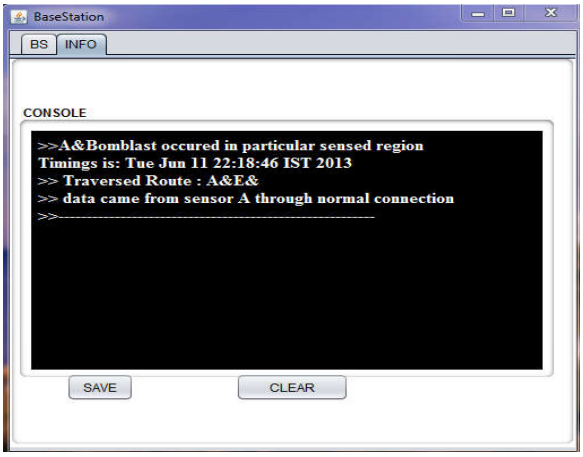

Fig 6.6 information stored in base station

In the above fig 8.6 finally the data came from sensor node A through normal connection. The message details are bomb blast occurred in particular sensed region.

Timing is: "Tue Jun 11 22:18:46 IST 2013, The base station returns acknowledgment to node A through node E".

Broadcasting energy cost report from sensor node A

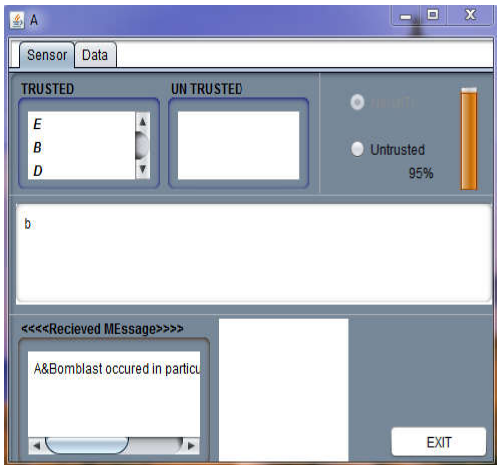

Fig 6.7 broadcasting process

For every transmission of data, each sensor node broadcasts its energy level to its neighbor nodes .In above fig 8.7 the node A broadcast its energy cost is 95% to its all neighbor nodes.

Broadcasting untrusted node E to all nodes

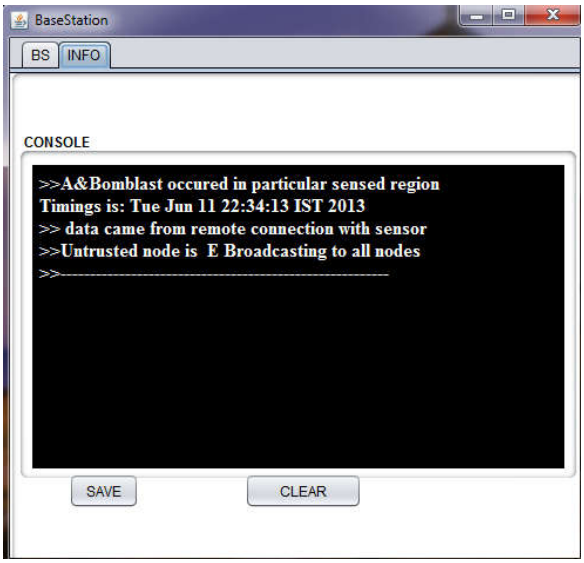

Fig 6.8 Broadcasting untrusted node

If suppose the base station finds any untrusted nodes in the network it considers them as a candidates, and broadcasts those nodes to all the sensor nodes which are present in the network. In above fig 8.8 base station finds E as the untrusted node and finally it broadcasting to all nodes.

Registration node

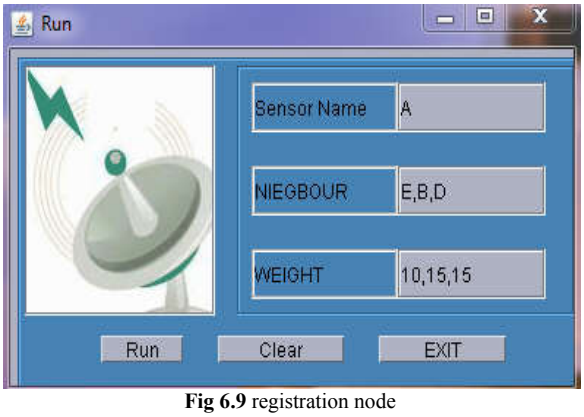

Fig 6.9 registration node

Registration node helps in registering its all sensor node; each and every node should be registered under this node. Registration node contains following details of each sensor node.

1. Sensor name
2. Node id
3. Public key of each node
4. Neighbors nodes of each sensor nodes

CONCLUSION

Finally in this paper presents a robust trust-aware routing framework for WSNs, to secure multi-hop routing in dynamic WSNs against harmful attackers exploiting the replay of routing information. TARF focuses on trust worthiness and energy efficiency, which are vital to the survival of a WSN in a hostile environment. With the idea of trust management, TARF enables a node to keep track of the trustworthiness of its neighbors and thus to select a reliable route. . The proposed routing solution was successfully implemented and validated in real-life sensor nodes proving its implementation feasibility. The involved implementation cost mainly depends on the adoption of a reputation exchange protocol and on the number of behavior aspects used for the evaluation of each node's trustworthiness. Unlike previous efforts at secure routing for WSNs, TARF effectively protects WSNs from severe attacks through replaying routing information; it requires neither tight time synchronization nor known geographic information. The resilience and scalability of TARF is proved through both extensive simulation and empirical evaluation with large-scale WSNs; the evaluation involves static and mobile settings, hostile network conditions, as well as strong attacks such as wormhole attacks and Sybil attacks. TARF module can be integrated into existing routing protocols with the least effort, thus producing secure and efficient fully-functional protocols. Finally, demonstrates a proof-of-concept mobile target detection application that is built on top of TARF and is resilient in the presence of an anti-detection mechanism that indicates the potential of TARF in WSN applications.

Future Scope

In this project presents challenges and design issues in implementing energy efficient routing protocols in wireless sensor networks, various kinds of attacks and its behavior. And lastly different algorithms that take into consideration the energy trust and distance matrices of the sensor network nodes. In future we can implement and do comparative study of these energy and trust aware algorithms.

References

1. G. Zhan, W. Shi, and J. Deng, "Tarf: A trust-aware routing framework for wireless sensor networks," in Proceeding of the 7th European Conference on Wireless Sensor Networks (EWSN'10), 2010.
2. A. Wood and J. Stankovic, "Denial of service in sensor networks," *Computer*, vol. 35, no. 10, Oct 2002. PP. 54–62.
3. A. Perrig, R. Szewczyk, W. Wen, D. Culler, and J. Tygar, "SPINS: Security protocols for sensor networks," *Wireless Networks Journal* (WINET), vol. 8, no. 5, Sep. 2002, pp. 521-534.
4. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," IEEE Communications Magazine, vol. 40, no. 8, Aug. 2002, pp. 102-114.
5. C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," in Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications, 2003.
6. Z. Yan, P. Zhang, and T. Virtanen, "Trust evaluation based security solution in ad hoc networks," in Proceeding of the 7th Nordic Workshop on Secure IT Systems, 2003.
7. F. Zhao and L. Guibas, Wireless Sensor Networks: An Information Processing Approach. Morgan Kaufmann Publishers, 2004.
8. J. Newsome, E. Shi, D. Song, and A. Perrig, "The sybil attack in sensor networks: Analysis and defenses," in Proc. of the 3rd International Conference on Information Processing in Sensor Networks (IPSN'04), Apr. 2004.