International Journal of Sensors and Sensor Networks
2013; 1(5): 61-68
Published online September 30, 2013 (http://www.sciencepublishinggroup.com/j/ijssn)
doi: 10.11648/j.ijssn.20130105.13

Energy efficient and trust metric based routing technique using collection tree protocol for WSNs

Pooja Kalidas Shinde*, Veeresh Gangappa Kasabegoudar

Post Graduate Department, Mahatma Basaveshwar Education Society's, College of Engineering, Ambajogai, India, 431 517

Email address:
Pooh26shinde@gmail.com (P. K. Shinde), veereshgk2002@rediffmail.com (V. G. Kasabegoudar)

To cite this article:
Pooja Kalidas Shinde, Veeresh Gangappa Kasabegoudar. Energy Efficient and Trust Metric Based Routing Technique Using Collection Tree Protocol for WSNs. *International Journal of Sensors and Sensor Networks.* Vol. 1, No. 5, 2013, pp. 61-68.
doi: 10.11648/j.ijssn.20130105.13

Abstract: In this paper, a readily deployable trust and energy-aware routing protocol is presented. A distributed trust management system incorporating direct and indirect trust information is used to detect and avoid malicious nodes performing routing attacks as well as attacks threatening the reputation exchange process. Also, the energy-awareness is relied upon to extend the network lifetime. Although, significant research effort has been spent on the design of trust models to detect malicious nodes based on direct and indirect evidence, this comes at the cost of additional energy consumption. In order to enhance the security of routing information between the nodes, energy efficient and trust metric based routing protocol using collection tree protocol (CTP) for wireless sensor networks (WSN) has been proposed. Simulated results presented here indicate that the proposed protocol satisfactorily performs the routing and is strong against attacks by exploiting the replay of routing information.

Keywords: WSN, Trust Model Management, Energy Efficient Routing, Secure Routing, Routing in WSN Survey

1. Introduction

Wireless Sensor Networks (WSN) refer to a group of spatially dispersed and dedicated sensors for monitoring and recording the physical condition of the environment and organizing the collected data at a central location. The sensor node equipment includes a radio transceiver along with an antenna, microcontroller, an interfacing electronic circuit, and an energy source usually a battery. In the proposed scheme, routing decisions are based on a weighted routing cost function which incorporates trust, energy and location attributes [1-4]. Number of routing protocols is available in literatures which work on use of efficient energy in order to provide security from malicious nodes. To select next intermediate node neighborhood table management is used to store information of each intermediate node [5]. There are several energy-aware routing techniques which have been proposed for wireless sensor networks. These routing techniques can be classified in to three categories: Data Centric, Hierarchical Cluster based, and Location Based Routing [2]. Many researchers have proposed several wireless ad hoc network protocols which can used in wireless sensor networks. Some of them are proactive like destination sequenced distance vector (DSDV) [6] and designed for static networks. Collection tree protocol is a tree based and whose main objective is to provide best effort any cast datagram communication to one of the collection roots node in the network [7].

Mobile agent technology provides an effective method to overcome the bottlenecks. Mobile agents transfer algorithms and process data locally, which decrease the demand for network bandwidth and reduce the capability of program codes [8-10]. Proposed work focuses on various kinds of attacks in which adversaries misdirect network traffic by identity deception through routing information [11-13].

Trust is evaluated by direct observation and second-hand information distributed among a network. In the latter category, trust in neighbors is evaluated by direct observation and trust relations between two nodes [14-19]. Each of these scattered sensor nodes has the capability to collect and route data either to other sensors or back to an external base stations (BS). A base station (BS) may be a fixed or a mobile node capable of connecting the sensor network to an existing communications infrastructure or to the internet where a user can have access to the reported data.

Unfortunately, most of the existing routing protocols for WSNs either focus on energy efficiency [20], assuming that

each node is honest with its identity, or they try to exclude unauthorized participation by encrypting data and authenticating packets. Examples of these encryption and authentication schemes for WSNs include Tinysec [21] and Spins [22] softwares. Also, it is important to consider efficient energy usage for battery-powered sensor nodes and the robustness of routing under topological changes and common faults in a wild environment. However, it is also significant to incorporate security as one of the most important goals; meanwhile, even with perfect encryption and authentication schemes, by replaying routing information, a malicious node can still participate in the network using another valid node's identity. The proposed protocol works on both energy efficiency and trust management. In contrast, trust management [17] has been introduced into peer-to-peer networks and general ad hoc networks to support decision-making, improve security, and promote node collaboration and resource sharing. Basically, trust management assigns each node a trust value according to its past performance. These studies target general ad hoc networks and peer-to-peer networks but not resource-constrained WSNs. Additionally, they do not address attacks arising from the replay of routing information.

The protocol proposed is efficient, robust, and reliable in a network with highly dynamic link topology. It quantifies link quality estimation in order to choose a next-hop node. This protocol creates a framework for CTP using trust metric. Unlike other security measures, proposed protocol neither requires tight time synchronization nor known geographic information. Most importantly, trust aware CTP proves resilient under various attacks exploiting the replay of routing information, which is not achieved by previous security protocols. Therefore, in this paper, a security routing protocol with the help of energy efficiency and trust metric is proposed to avoid different types of attacks and also elaborate the challenges and design issues in order to overcome the future difficulties.

Section 2 explains the design of TARF into existing protocol in order to bring more security in proposed routing protocol. In Section 3, the empirical evaluation of large sensor network with routing procedure (EnergyWatcher and TrustManager) is explained. Section 4 presents the simulation results of TARF against various attacks through routing information in static and mobile conditions. Finally, Section 5 explains the conclusions and future scope of the work presented in this paper.

2. Integration of TARF into Existing Protocol

To demonstrate the working of trust metric based CTP, we incorporated TARF into collection tree routing protocol. Similar to the original CTP's implementation, the implementation of this new protocol decides the next-hop neighbor for a node with two steps (pl. ref. Section 3.4). Neighborhood table is traversed for an optimal candidate for the next hop whereas Step 2 decides to switch from the current next-hop node to the optimal candidate found in step 1. For Step 1, as in the CTP implementation, a node would not consider those links congested, likely to cause a loop, or having a poor quality lower than a certain threshold. This new implementation prefers those candidates with highest trust levels; in certain circumstances, regardless of the link quality, the rules deem a neighbor with a much higher trust level to be a better candidate. The preference of highly trustable candidates is based on the following consideration: On the one hand, it creates least chance for an adversary to misguide other nodes into a wrong routing path by forging the identity of an attractive node such as a root; on the other hand, forwarding data packets to a candidate with a low trust level would result in many unsuccessful link level transmission attempts, thus leading to much retransmission and a potential waste of energy. When the network throughput becomes low and a node has a list of low trust neighbors for routing decisions. As for step 2, compared to the CTP implementation, two more circumstances are added when a node decides to switch to the optional candidate found in step 1 that the candidate has a higher trust level, or the current next–hop neighbor has a too low trust level [1]. As shown in Figure 1 each node selects a next-hop node based on its neighbourhood table, and broadcast its energy cost within its neighbourhood. To maintain this neighbourhood table, EnergyWatcher and TrustManager on the node keep track of related events (on the left) to record the energy cost and the trust level values of its neighbours.

2.1. Proposed Protocol

2.1.1. Design Consideration

Before elaborating the detailed design of TARF, it is essential to clarify a few design considerations first, including certain assumptions. It may be noted that only one base station is used for the implementation. Additionally, to merely simply the introduction of TARF, it is assumed that no data aggregation is involved. It is also assumed that the data packet has at least the following fields: sender id, the sender sequence no, the next-hop node id (the receiver in this one hop transmission), the sourceid (the node that initiates the data), and the source sequence no. Also, the source nodes information should be included for the reasons that the base station has to track whether a data packet is delivered or not.

2.1.2. Goals

High Throughput: Throughput is defined as the ratio of the number of all data packets delivered to the base station to the number of all sampled data packets. In our evaluation, throughput at a moment is computed over the period from the beginning time (0) until that particular moment.

Energy Efficiency: Data transmission accounts for a major portion of the energy consumption. We evaluate

energy efficiency by the average energy cost to successfully

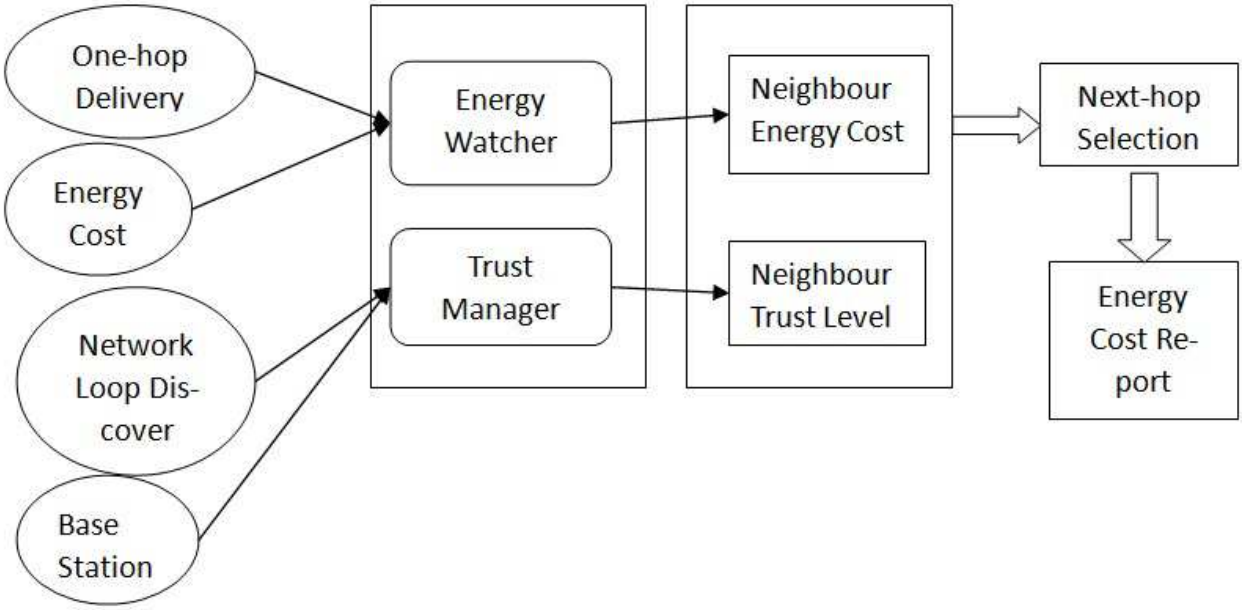

Figure 1: *Block diagram of TARF [1].*

deliver a unit-sized data packet from a source node to the base station. The energy consumption depends on the number of hops, i.e., the number of one-hop transmissions occurring. To evaluate how efficiently energy is used, we can measure the average hops that each delivery of a data packet takes, abbreviated as hop-per-delivery.

Scalability and Adaptability: TARF should work well with WSNs of large magnitude under highly dynamic contexts. We will evaluate the scalability and adaptability of TARF through experiments with large-scale WSNs and under mobile and hash network conditions.

2.1.3. Challenges and Design Issues

The design of energy-efficient routing protocols in WSNs is influenced by many factors. These factors must get over before efficient communication can be achieved in WSNs. Here is a list of the most common factors affecting the routing protocols design [2].

- Node Deployment, Coverage, Quality of Service, Data Aggregation
- Faulty/ Fraudulent Nodes

Some nodes in WSN may drop the data or send incorrect data. It is needful to detect such nodes to reduce data loss. Some approaches are also defined to calculate trust level of the nodes. Table 1 represents various types of attacks occur during routing the packet.

3. Proposed TARF

TARF secures the multi-hop routing in WSNs against intruders misdirecting the multi-hop routing by evaluating the trustworthiness of neighboring nodes. It identifies such intruders by their low trustworthiness and routes data through paths circumventing those intruders to achieve satisfactory throughput. TARF is also energy efficient, highly scalable, and well adaptable.

- Neighbor (N): For a node N which may be source or sender, neighbor (neighboring node) of N is a node that is reachable from N with one-hop wireless transmission.
- Trust Level Metric (T): For a node N, the trust level of a neighbor is a decimal number in [0, 1], representing N's opinion of that neighbor's level of trustworthiness. Specifically, the trust level of the neighbor is N's estimation of the probability that this neighbor correctly delivers data received to the base station.
- Energy Cost (E): For a node N, the energy cost of a neighbor is the average energy cost to successfully deliver a unit sized data packet with this neighbor as its next-hop node, from N to the base station.

3.1. Routing Procedure

TARF, as with many other routing protocols, runs as a periodic service. 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 a field 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. To maintain the stability of its routing path, a node may retain the same

Table 1: *Network layer attacks and behavior*

Attack type	Attack type
Selfish behavior(black hole, grey hole)	A malicious node denies performing benign routing and drops part or the entire received packet.
Sinkhole attack	A malicious node tries to attract traffic advertising fake routing information, and then it refuses to forward it.
Replay attack	The original routing messages are repeated at a later time, thus deceiving the routing functionality.
Modification attack	An adversary modifies the data and/or routing packets it forwards
Sybil attack	An attacker presents multiple identities.
Traffic analysis attack	A malicious node monitors the traffic flows in order to identify, locate and attack the critical nodes (typically the base station).

next-hop node until the next fresh broadcast message from the base station occurs [1]. Next, we introduce the structure and exchange of routing information as well as how nodes make routing decisions in TARF.

3.2. Structure and Exchange of Routing Information

A broadcast message of delivery of packets consists of small no of packets. Each such packet consist of <node id of a source node, an undelivered sequence interval [a, b] with a significant length>, <node id of a source node, minimal sequence number received in last period, maximum sequence number received in last period>, as well as several node id intervals of those without any delivery record in last period.

Roughly, the effectiveness can be explained as follows: the fact that an attacker attracts a great deal of traffic from many nodes often gets revealed by at least several of those nodes being deceived with a high likelihood. The undelivered sequence interval [a, b] is explained in [1]. Accordingly, each node in the network stores a table of <node id of a source node, a forwarded sequence interval [a, b] with a significant length> about last period. The data packets with the source node and the sequence numbers falling in this forwarded sequence interval [a, b] have already been forwarded by this node. When the node receives a broadcast message about data delivery, its TrustManager will be able to identify which data packets forwarded by this node are not delivered to the base station. Considering the overhead to store such a table, old entries will be deleted once the table is full [1].

3.3. Energy Watcher

Here, how a node N's EnergyWatcher computes the energy cost E_{Nb} for its neighbour b in N's neighbourhood table and how N decides its own energy cost E_N is explained. Also, one-hop retransmission may occur until the acknowledgment is received or the number of retransmissions reaches a certain threshold. It may be noted that the retransmission cost needs to be considered. With the above notations, it is straightforward to establish the following relation [1].

$$\mathrm{E_{Nb}} = E_{N\to b} + E_b \tag{1}$$

Where $E_{N\to b}$ represents the energy from node N to b and is equal to the ratio of E_{unit} to the probability of success of a node [1]. Then, we have

$$E_{Nb} = E_{unit}/P_{succ} + E_b \tag{2}$$

The purpose of E_{Nb} is to get the probability p_{succ} that a one-hop transmission is acknowledged. Considering the variable wireless connection among wireless sensor nodes, we do not use the simplistic averaging method to compute p_{succ}. Instead, after each transmission from N to b, N's EnergyWatcher will update p_{succ} based on whether that transmission is acknowledged or not with a weighted averaging technique. We use a binary variable Ack to record the result of current transmission: 1 if an acknowledgment is received; otherwise, 0. Given Ack and the last probability value of an acknowledged transmission p_{old_succ}, an intuitive way is to use a simply weighted average of Ack and p_{old_succ} as the value of p_{new_succ}. That is what is essentially adopted in the aging mechanism [3]. However, that method used against sleeper attacks still suffers periodic attacks [4]. To solve this problem, p_{succ} is updated with the value using two different weights as suggested in [4], a relatively big $w_{degrade} \in (0, 1)$ and a relatively small $w_{upgrade} \in (0, 1)$ as follows [1]:

$$p_{new\ succ} = \begin{cases} (1 - w_{degrade}) \times p_{old\ succ} + w_{degrade} \times Ack, \\ \qquad if\ Ack = 0 \\ (1 - w_{upgrade}) \times p_{old\ succ} + w_{upgrade} \times Ack, \\ \qquad if\ Ack = 1 \end{cases} \tag{3}$$

3.4. Trust Manager

A node N's TrustManager decides the trust level of each neighbour based on the following events: discovery of network loops, and broadcast from the base station about data delivery. For each neighbour b of N, T_{Nb} denotes the trust level of b in N's neighbourhood table. At the beginning, each neighbour is given a neutral trust level 0.5. After any of those events occurs, the relevant neighbours' trust levels are updated. Note that many existing routing protocols have their own mechanisms to detect routing loops and to react accordingly [5-7]. In that case, when integrating TARF into those protocols with antiloop mechanisms, TrustManager may solely depend on the broadcast from the base station to decide the trust level; we adopted such a policy when implementing TARF later (see Section 3.5).

Though sophisticated loop-discovery methods exist in the currently developed protocols, they often rely on the comparison of specific routing cost to reject routes likely leading to loops [7]. To minimize the effort to integrate

TARF and the existing protocol and to reduce the overhead, we adopt the following mechanism to detect routing loops [1]. We use a binary variable Loop to record the result of loop discovery: 0 if a loop is received; 1 otherwise. As in the update of energy cost, the new trust level of b is [1]:

$$T_{new\ Nb}= \begin{cases} (1-w_{degrade}) \times T_{old\ Nb} + w_{degrade} \times Loop, \\ \qquad if\ Loop = 0 \\ (1-w_{upgrade}) \times T_{old\ Nb} + w_{upgrade} \times Loop, \\ \qquad if\ Loop = 1 \end{cases} \quad (4)$$

Once a loop has been detected by N for a few times so that the trust level of the next-hop node is too low, N will change its next-hop selection, thus that loop is broken [1]. It computes the ratio of the number of successfully delivered packets which are forwarded by this node to the number of those forwarded data packets, denoted as Delivery Ratio. Then, N's TrustManager updates its next hop node b's trust level as follows [1]:

$$T_{new\ Nb}= \begin{cases} (1-w_{degrade}) \times T_{old\ Nb} \\ +w_{degrade} \times DeliveryRatio, \\ if\ DeliveryRatio < T_{\ old\ Nb} \\ (1-w_{upgrade}) \times T_{old\ Nb} \\ +w_{upgrade} \times DeliveryRatio, \\ if\ DeliveryRatio \geq T_{old\ Nb} \end{cases} \quad (5)$$

3.5. Routing Decision with Trust Management

The algorithm presented here [1] gives the idea of working of neighbourhood table to select the next optimal candidate. Neighbourhood table helps to select most trusted candidate to route the packet to base station in order to avoid malicious node from entering in the network. The algorithm also avoids the loops in network.

```
//Step 1: traverse the neighbourhood table for an optimal candidate for the next hop
optimal_candidate = NULL
//the cost of routing via the optimal candidate provided by the existing protocol, initially infinity
optimal_cost = MAX_COST
//the trust level of the optimal candidate, initially 0
optimal_trust = MIN_TRUST
for each candidate in the neighbourhood table
if the link is congested, or may cause a loop, or does not pass quality threshold
Continue

better = false
if candidate.trust >= optimal_trust && candidate.cost < optimal_cost
better = true

//prefer trustworthy candidates
if candidate.trust >= TRUST_THRESHOLD && optimal_trust < TRUST_THRESHOLD
better = true
if candidate.trust >= ESSENTIAL_DIFFERENCE_THRESHOLD + optimal_trust
better = true

//effective when all nodes have low trust due to network change or poor connectivity
if candidate.trust >= 3*optimal_trust/2
better = true

//add restriction of trust level requirement
if candidate.trust >= TRUST_THRESHOLD && candidate.trust/candidate.cost > optimal_trust/optimal_cost
better = true
if better == true
optimal_candidate=candidate
optimal_cost=candidate.cost
optimal_trust = candidate.trust

//Step 2: Decide whether to switch from the current next-hop node to the optimal candidate found

if optimal_trust>= currentNextHop.trust
|| currentNextHop.trust<=TRUST_THRESHOLD
|| current link is congested and switching is not likely to cause loops
|| optimal_cost + NEXTHOP_SWITCH_THRESHOLD < currentNextHop.Cost
CurrentNextHop = optimal_candidate.
```

4. Simulation Results

The proposed protocol is implemented in Network Simulator (NS2) software. The parameters used for comparison between fixed node CTP and movable node CTP are packet delivery ratio, throughput, delay, jitter, control overhead, and average energy. All these parameters have been investigated against sense time and with & without attacks versus simulation time. All these parameters are analyzed with network simulator and their performances are presented in Figures 2 to 7 and the analysis of all these figures is summarized in Table 2. Simulation parameters used are listed in Table 3 for quick reference.

Table 2: *Performance analysis of different parameters*

Parameter	PDR	Throughput	Delay (ms)	Jitter (ms)	Control overhead	Average energy (J)
Sense time	Decreases	Decreases	Decreases	Increases	Decreases	Decreases
With attack	Decreases	Increases	Increases	Nearly constant	Increases	Increases
Without attack	Increases	Nearly constant	Increases	Decreases	Same as with attack	Same as with attack

Table 3: *Simulation parameters*

Parameters	Value
Source Type	MAC
No. of Nodes	50, 75, 100, 125, 150, 175
Simulation Time	140 sec
Environmental Size	100*100
Transmission Range	100 m
Traffic size	Constant Bit Rate(CBR)
Packet Size	512
Packet Rate	5 packets/sec
Maximum Speed	20 m/s
Pause Time	5ms

Following points may be noted from Figures 2 to 7 and Table 3:

- When the speed of node increases PDR_sensetime and PDR with attack decrease whereas PDR without attack increases (Figure 2).
- Similarly, when the speed of a node increases throughput sense time goes down and throughput without attack increases as applied to trust metric (Figure 3).
- The performance of protocol is compared between average End-to-End delay and simulation time along with presence of traffic nodes 50, 75, and 100 with varying number of simulation time i.e. 60, 80, and 100 Sec. in the network. The simulated values of average end-to-end delay represent that reliability of routing protocol in the network. As we go away from sensing range the delay_sensetime decreases. It happens because sense time is not in present area (Figure 4).
- Jitter is reduced with time as it is free from attacks. Jitter is employed to avoid collisions caused by simultaneous transmission by adjacent nodes over the same channel. As the number of packets increased over the same channel, jitter is increased which leads to the loss of data (Figure 5).
- Figure 6 represents the normalized control overhead which increases with time even though the number of nodes increases. This ratio is calculated by comparing the total number of routing packets transmitted during the simulation time to the number of data packets delivered.
- When the node speed increases with time average energy increases as it is fully trusted. From the plots (Figures 2 & 6) it is observed that lower the speed of node higher the packet delivery ratio, because increase in speed of mobile node will increases the probability of breaking the routes.

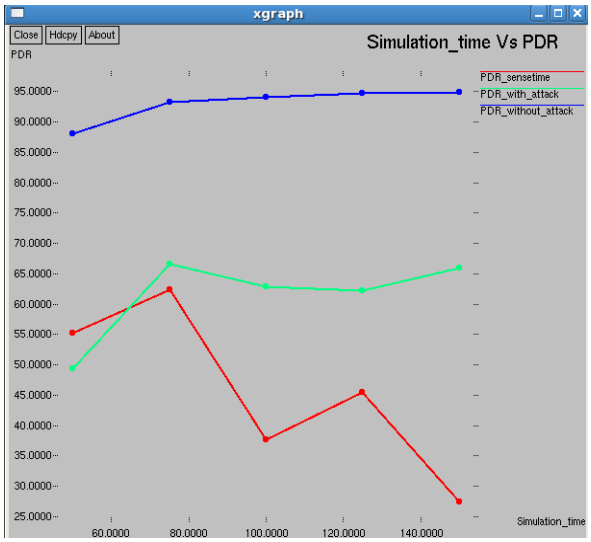

Figure 2: *Simulation time vs. PDR*

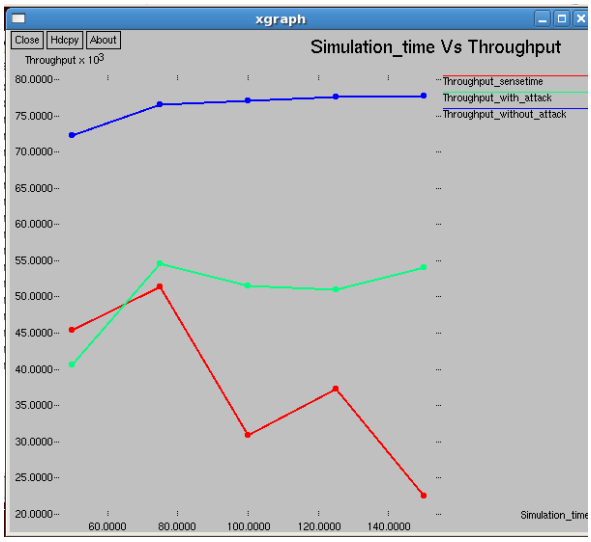

Figure 3*: Simulation time vs. throughput*

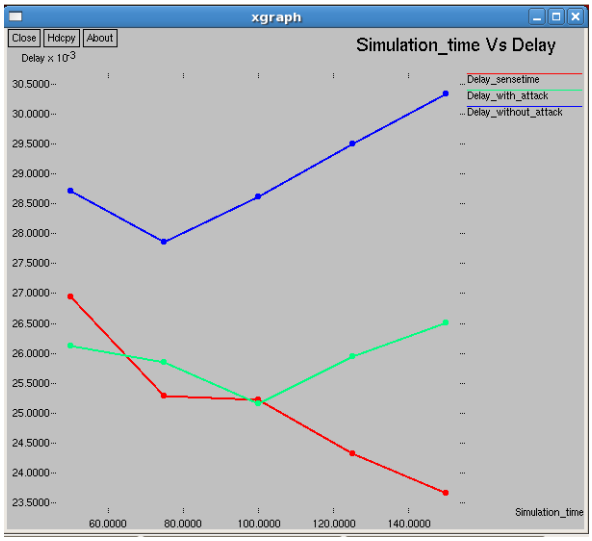

Figure 4: *Simulation time vs. delay*

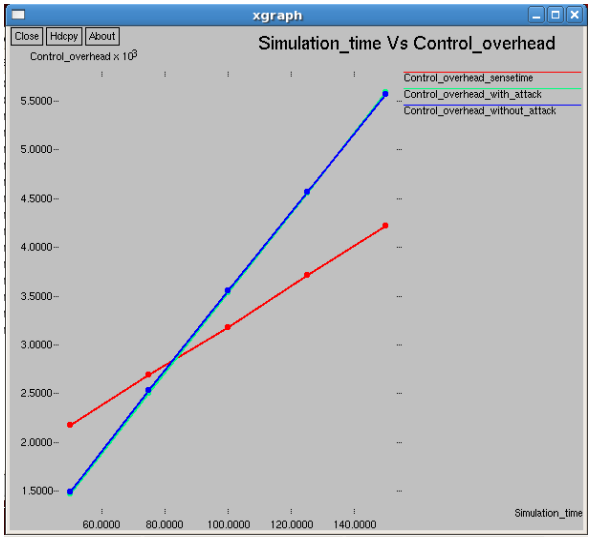

Figure 6: *Simulation time vs. control overhead*

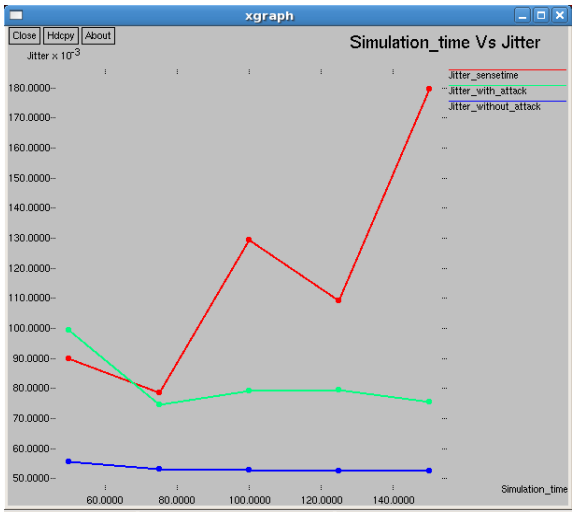

Figure 5: *Simulation time vs. jitter*

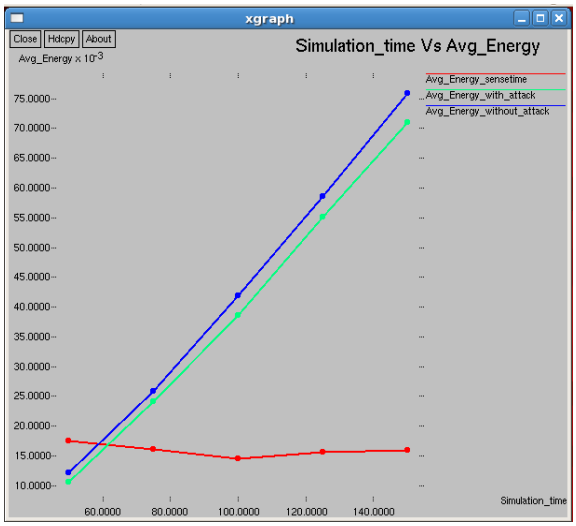

Figure 7: *Simulation time vs. average energy*

5. Conclusions

Collecting data at a base station is a common requirement of sensor network applications. The general approach used is to build one or more collection trees, each of which is rooted at a base station. When a node has data which needs to be collected, it sends the data up the tree, and it forwards collection data that other nodes send to it. Not only does TARF circumvent those malicious nodes misusing other nodes' identities to misdirect network traffic, it also accomplishes efficient energy usage. Simulation results presented here indicate that:

- Efficiency of energy usage in TARF is generally at least comparable to that in existing protocols.
- With the existence of traffic misdirection through "identity theft", TARF generally achieves a significantly higher throughput than other existing protocols. And,
- TARF is scalable and adaptable to typical medium-scale test bed environments and simulated conditions.

The future work will address the fact that when the number of isolated malicious nodes increases, some nodes may find them totally surrounded by malicious neighbours and cannot participate effectively in the network. Several mechanisms may be used to solve this issue.

One possible solution can be making the nodes that are totally surrounded by malicious neighbours adjust dynamically their belief and disbelief thresholds. Another solution is to give malicious nodes a chance to repent, by letting them broadcast repent packet to their 1-hop neighbours, which can place them on a probation period before deciding whether to forgive them or not.

References

[1] G. Zhan, W. Shi, and J. Deng, "Design and implementation of tarf: a trust aware routing framework for wsns," *IEEE Transaction on Dependable and Secure Computing*, vol. 9, no. 2, pp.184-197, 2012.

[2] F. Zhao and L. Guibas, Wireless Sensor Networks: Information Processing Approach, Elsevier Morgan Kaufmann, Boston, 2004.

[3] S. Ganeriwal, L. Balzano, and M. Srivastava, "Reputation-based framework for high integrity sensor networks," *ACM Trans. Sensor Networks*, vol. 4, pp. 1-37, 2008.

[4] G. Zhan, W. Shi, and J. Deng, "Poster abstract: sensortrust—a resilient trust model for WSNs," *Proc. Seventh Int'l Conf. Embedded Networked Sensor Systems (SenSys '09)*, pp. 1-3, 2009.

[5] A. Woo, T. Tong, and D. Culler, "Taming the underlying challenges of reliable multihop routing in sensor networks," *Proc. First ACM Int'l Conf. Embedded Networked Sensor Systems*, pp. 14-27, 2003.

[6] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers," *ACM SIGCOMM Computer Comm. Rev.,* vol. 24, no. 4, pp. 234-244, 1994.

[7] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection tree protocol," *Proc. Seventh ACM Conf. Embedded Networked Sensor Systems*, pp. 1-14, 2009.

[8] L. Bai, F. Ferrese, K. Ploskina, and S. Biswas, "Performance analysis of mobile agent-based wireless sensor network," *Proc. Eighth Int'l Conf. Reliability, Maintainability and Safety*, pp. 16-19, 2009.

[9] L. Zhang, Q. Wang, and X. Shu, "A mobile-agent-based middleware for wireless sensor networks data fusion," *Proc. Instrumentation and Measurement Technology Conf. (I2MTC '09)*, pp. 378-383, 2009.

[10] W. Xue, J. Aiguo, and W. Sheng, "Mobile agent based moving target methods in wireless sensor networks," *Proc. IEEE Int'l Symp. Comm. and Information Technology (ISCIT '05)*, vol. 1, pp. 22- 26, 2005.

[11] A. Wood and J. Stankovic, "Denial of service in sensor networks," *Computer*, vol. 35, no. 10, pp. 54-62, 2002.

[12] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," *Proc. First IEEE Int'l Workshop Sensor Network Protocols and Applications,* pp. 113-127, 2003.

[13] J. Newsome, E. Shi, D. Song, and A. Perrig, "The sybil attack in sensor networks: analysis and defences," *Proc. Third Int'l Conf. Information Processing in Sensor Networks (IPSN '04)*, pp. 259-268, 2004.

[14] G.Theodorakopoulos and J. S. Baras , " On trust models and trust evaluation metrics for ad-Hoc networks," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 24, no. 2, pp. 318-328, 2006.

[15] C. Liu, Y. Liu and Z. Zhang, "Improved reliable trust-based and energy-efficient data aggregation for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, pp. 1-11, 2013.

[16] I. Rijin, N. Sakthivel, and S. Subasree, "Development of an enhanced efficient secured multihop routing technique for wsns," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 1, no. 3, pp. 506-512, 2013.

[17] P. Deepa, A. Shalini, and J. Joshi, "Trust management for mobile ad hoc networks using recommendation exchange protocol," *International Journal of Computer Trends and Technology*, vol.1, no. 2, pp. 115-118, 2011.

[18] G. Crossby, L. Hester, and N. Pissinou, "Location-aware, trust-based detection and isolation of compromised nodes in wsns," *International Journal of Network Security*, vol. 12, no.2, pp. 107-117, 2011.

[19] H. Chen, "Task based trust management for wsns," *International Journal of Security and its Applications*, vol. 3, no. 2, pp. 22-26 2009.

[20] J. Al-Karaki and A. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE Wireless Communications*, vol. 11, no.6, pp. 6–28, 2004.

[21] R. Perrig, Szewczyk, W. Wen, D. Culler, and J. Tygar, "Security protocols for sensor networks(SPINS)," *Wireless Networks Journal (WINET)*, vol. 8, no.5, pp. 521–534, 2002.

[22] C. Karlof, N. Sastry, and D. Wagner, "Tinysec: link layer security architecture for wireless sensor networks," *In. Proc. of ACM SenSys*, pp.162-175, 2004.