International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol. 3, Issue. 5, Sep - Oct. 2013 pp-2867-2871 ISSN: 2249-6645

Performance Analysis of Trust-Aware Routing Framework for Wireless Mesh Networks

B. Sai Pragna [1], M. Shakeel Ahmed[2,] B.Sai Manogna[3]
#1,3 M.TECH Scholar, PBRVITS, Kavali
#2 Associate Professor, PBRVITS, Kavali

ABSTRACT: *Multi-hop routing in wireless sensor networks (WSNs) offer small protection against trickery throughout replaying routing information. A challenger can develop this defect to launch various harmful or even devastating attacks against the routing protocols, including wormhole attacks, sinkhole attacks and Sybil attacks. Even though important research effort has been spend on the design of trust models to detect malicious nodes based on direct and indirect confirmation, this comes at the cost of extra energy consumption. Conventional cryptographic techniques or efforts at mounting trust-aware routing protocols do not effectively address these problems. To secure the wireless sensor networks against adversaries misdirecting the multi-hop routing, we have preposed TARF, a robust trust-aware routing framework for dynamic WSNs. Without prolonged time synchronization or known geographic information, TARF offers dependable and energy-efficient route. TARF demonstrates effective adjacent to those harmful attacks developed out of identity trickery; the flexibility of TARF is verified through extensive assessment with both simulation and observed experiments on large-scale WSNs under various scenarios including mobile and RF-shielding network conditions. We have put into action a low-overhead TARF module in TinyOS; this implementation can be included into existing routing protocols with the least effort. Based on TARF, we also verified a proof-of-concept mobile target detection application that functions well next to an anti-detection mechanism.*

Index Terms: *Wireless Sensor Networks, Wireless Sensor Network, Trusted Aware Routing Framework (TARF), Congestio,. TinyOS*

I. INTRODUCTION

Wireless sensor networks (WSNs) are models for applications to report detected events of interest, such as forest fire monitoring and military surveillance. A Wireless sensor networks includes battery powered senor nodes with exceptionally limited processing abilities. With a narrow radio communication range, a sensor node wirelessly passes messages to a base station via a multi-hop path. Though, the multi-hop routing of WSNs often becomes the target of wicked attacks. An attacker may interfere nodes actually, create traffic collision with apparently valid transmission, fall or misdirect communication in routes or jam the communication channel by creating radio interference. As a risky and easy-to-implement type of attack, a malicious node basically replays all the outgoing routing packets from a applicable node to copy the latter node's uniqueness; the malicious node then uses this fake identity to contribute in the network routing, thus trouble making the network traffic. Even if malicious node cannot straightly listen in the valid node's wireless transmission, it can scheme with other malicious nodes to receive those routing packets, which is identified as a wormhole attack. A node in a Wireless sensor networks relies exclusively on the packets received to know about the sender's identity, replaying routing packets permits the wicked node to forge the individuality of this valid node. After "stealing" that valid characteristics, this malicious node is able to misdirect the network traffic. It may fall packets received, forward packets to another node not supposed toward be in the routing path, or form a transmission loop from side to side which packets are passed among a few malicious nodes infinitely.

Sinkhole attacks can be start on after thefting a applicable identity, in which a malicious node may maintain itself to be a base station through replaying all the packets from a genuine base station. Such a fake base station could attract more than half the traffic, creating a "black hole.". This technique can be engaged to conduct another strong form of attack Sybil attack: all the way through replaying the routing information of multiple legal nodes, an attacker may nearby multiple identities to the network. A valid node, if cooperated, can also launch all these attacks.

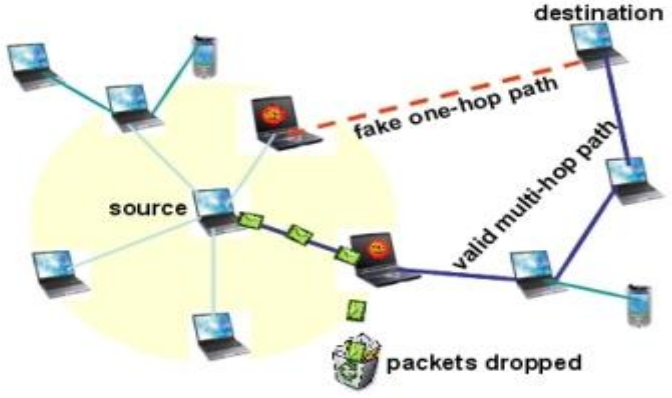


Fig.1: Attacks in multi hop routinng

Time synchronization service in wireless sensor networks has to meet challenges which are substantially different from those in infrastructure based networks. For example each sensor has a finite battery source and communication is expensive in terms of energy, an important issue of wireless sensor networks is energy efficiency. In addition, wireless sensor networks show a higher failure probability over the time than in traditional networks due to battery depletion or destruction of the sensors, and changes in the environment can severely affect radio propagation causing frequent network topology changes and network partitions. Moreover, at high densities wireless sensor networks become much more likely to suffer communication failures due to contention for their shared communication medium. These elements lead to robustness, strong energy efficiency and self configuration requirements. In the last several years, clock synchronization protocols for wireless sensor networks have been proposed based on different approaches ie., Reference Broadcast Synchronization (RBS) or hierarchical approaches, or interval based, or probabilistic approaches for energy efficiency. However, in spite of their diversity, these applications share a common viewpoint: they provide an accurate time estimate by means of periodic synchronization performed by each sensor node and based on messages exchanged with its neighbor nodes. Each clock adjustment is energy consuming since it involves transmitting messages and pay attention, besides the computational cost.

II. DESIGN CONSIDERATIONS

2.1. Assumptions

In this objective is secure routing for data collection tasks, which are one of the mainly fundamental functions of wireless sensor networks. In a data compilation task, a sensor node sends its example data to a remote base station with the help of other intermediate nodes, then there could be more than one base station, the direction-finding approach is not affected by the number of base stations that there is only one base station. An opponent may fake the identity of any legal node through replaying that node's outgoing routing packets and spoofing the acknowledgement packets, even remotely through a wormhole. In addition, to merely simplify the introduction of TARF to assume no data aggregation is involved.

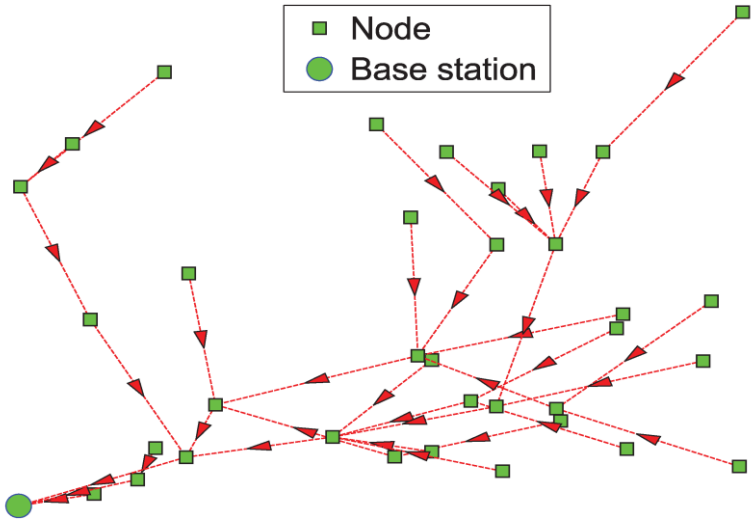


Fig. 2. Multi-hop routing for data collection of a WSN.

It is to be applied to cluster based wireless sensor networkss with static clusters, where data are cumulatively by clusters before being relayed. Cluster-based wireless sensor networkss allows for the great savings of energy and bandwidth through aggregating data from children nodes and performing routing and transmission for children nodes. In a cluster-based wireless sensor networks, the cluster headers themselves form a sub-network; after certain data arrive at a cluster header, the aggregated information will be routed to a base station only through such a sub network consisting of the cluster headers. The framework can be functional to this sub-network to achieve secure routing for cluster based wireless sensor networkss. TARF may run on cluster headers only and the cluster headers communicate with their children nodes directly since a static cluster has known relationship between a cluster header and its child nodes, even if any link-level security features may be further employed.

2.2. Authentication Requirements

Though a specific application may determine whether data encryption is needed, TARF requires that the packets are correctly authenticated, particularly the broadcast packets from the base station. The transmition from the base station is unevenly authenticated so as to guarantee that an adversary is not able to manipulate or forge a broadcast message from the base station at will. With authenticated broadcast, even with the existence of attackers, TARF may use TrustManager and the received broadcast packets about delivery information to choose trustworthy path by circumventing compromised nodes. Without being able to capturing the base station, it is generally very difficult for the opposition to manipulate the base broadcast packets from the base station is critical to any successful secure routing protocol. It can be achieved through existing irregularly authenticated broadcast schemes that may require loose time synchronization. As an example, µTESLA achieves asymmetric authenticated broadcast through a symmetric cryptographic algorithm and a loose delay schedule to disclose the keys from a key chain.

III. DESIGN OF TARF

TARF secures the multi-hop routing in wireless sensor networkss against intruders developing the repetation of routing information by evaluating the trustworthiness of neighboring nodes. It recognizes such intruders that misdirect obvious network traffic by their low trust advantage and routes data through paths circumventing those intruder to achieve reasonable throughput. TARF is also energy-efficient, highly scalable, and well flexible. Before introducing the detailed design, we initially introduce several essential notions here.

Neighbor: For a node*N*, a neighbor (neighboring node) of *N* is a node that is reachable from *N* with one-hop wireless transmission.
Trust level: 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. Particularly, the trust level of the neighbor is *N*'s estimation of the probability that this neighbor correctly delivers data received to the base station. That trust level is indicates as *T*.
Energy cost: 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. This energy cost is indicated as *E*.

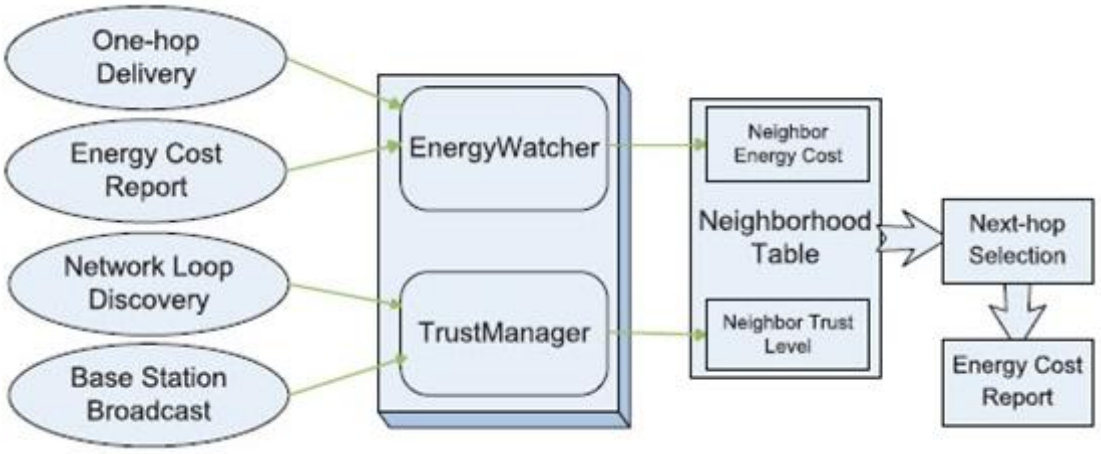


Fig. 3: Each node selects a next-hop node based on its neighborhood table, and broadcast its energy cost within its neighborhood. To maintain this neighborhood table, Energy- Watcher and TrustManager on the node keep track of related events (on the left) to record the energy cost and the trust level values of its neighbors.

3.2. Routing Procedure

TARF with as many other routing protocols, runs as a interrupted service. The length of that phase determines how regularly routing information is exchanged and reorganized. At the beginning of each period, the base station broadcasts a message regarding data release during last period to the whole network consisting of a few contiguous packets. Each such packet has a field to indicate how many packets are remaining to complete the broadcast of the current message. The achievement of the base station broadcast triggers the exchange of energy report in this new period. henever a node receives such a broadcast message from the base station, it recognizes that the most recent period has ended and a new period has just started. No fixed 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 TrustManager also keeps track of network loops and processes broadcast messages from the base station about data delivery to maintain trust level entries in its locality table.

3.3. Energy Watcher & Trust Manager

In this module Cluster-based wireless sensor networks allows for the great savings of energy and bandwidth through aggregating data from children nodes and performing routing and transmission for children nodes. In a cluster-based wireless sensor network, the cluster headers themselves form a sub network, after certain information appear at a cluster header, the collective data will be routed to a base station only through such a sub network consisting of the cluster headers. Framework can then be applied to this sub-network to achieve secure routing for cluster based wireless sensor networks. A node N's Trust Manager decides the trust level of each neighbor based on the following events: broadcast from the base station about data delivery and discovery of network loops. For each neighbor b of N, TNb denotes the trust level of b in N's neighborhood table. At the opening, each neighbor is given a neutral trust level 0.5. After any of those actions takes place, the relevant neighbors' trust levels are updated. Though sophisticated loop-discovery methods exist in the presently developed protocols, they often rely on the evaluation of detailed routing cost to reject routes likely most important to loops. To minimize the attempt to put together TARF and the existing protocol and to reduce the transparency, when an existing routing protocol does not offer any anti loop mechanism, it adopts the Probabilistic Clock Reading Method to detect routing loops.

IV. IMPLEMENTATION AND EMPIRICAL EVALUATION

In order to estimate TARF in a real-world setting, we execute the TrustManager component on TinyOS 2.x, which can be included into the existing routing protocols for wireless sensor networks with the least attempt. We implemented TARF as a self-contained routing protocol on TinyOS 1.x before this second implementation.

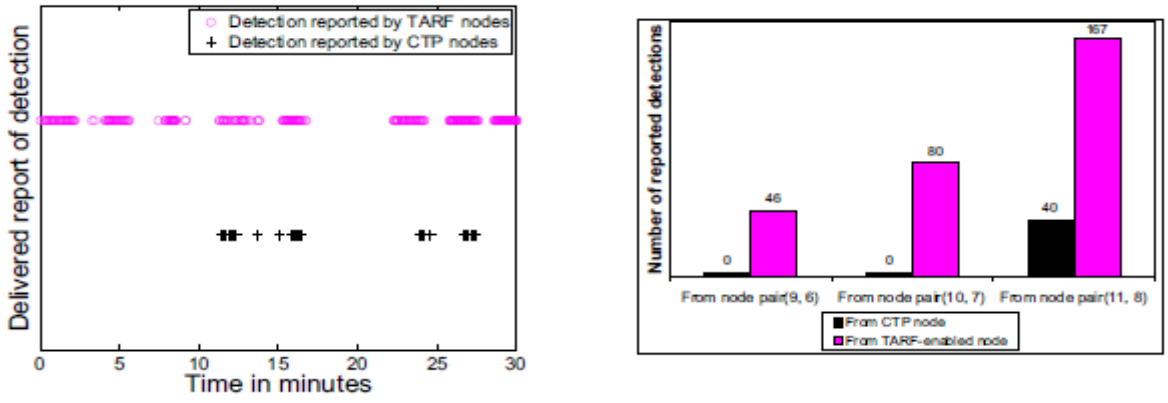
```
//Step 1. traverse the neighborhood table for an optimal candidate for the next hop
optimal_candidate = NULL
//the cost of routing via the optimal candidate provided by the existing protocol, initially infinity
optimal_cost = MAX_COST
//the trust level of the optimal candidate, initially 0
optimal_trust = MIN_TRUST
for each candidate in the neighborhood table
    if 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
```

Fig.4: Routing decision incorporating trust management.

1.1. TrustManager Implementation Details

The *TrustManager* component in TARF is enfolded into an self-determining TinyOS configuration named TrustManagerC. TrustManagerC uses a enthusiastic logic channel for communication and runs as a periodic check with a configurable period, thus not interfere with the application code. Although it is possible to implement TARF with a period always synchronized with the routing protocol's period that would cause much intrusion into the source code of the routing protocol. The current TrustManagerC utilizes a period of 30 seconds; for exact applications, by adjusting a convinced header file, the period extent may be re-configured to reflect the sensing occurrence, the energy effectiveness and trustworthiness requirement.

This new implementation integrating TARF requires moderate program storage and memory utilization. Here implemented a typical TinyOS data collection application, Multihop Oscilloscope, based on this new protocol. The Multihop Oscilloscope application, with certain modified sensing parameters for our later evaluation purpose, sometimes makes sensing samples and sends out the sensed data to a root via multiple routing hops. Originally, Multihop Oscilloscope uses CTP as its routing protocol. Now list the ROM size and RAM size necessity of both implementation of Multihop Oscilloscope on non-root Telosb motes in Table 1. The enabling of TARF in Multihop Oscilloscope increases the size of ROM by around 1.3KB and the amount of memory by around 1.2KB.



(a) Detection report. (b) Number of reported detections.

Fig. 5. Comparison of CTP and the TARF-enabled CTP in detecting the moving target.

V. CONCLUSIONS AND FUTURE WORK

We have designed and implemented a working model which is an enhanced version of TARF, a robust trust-aware routing framework for wireless sensor networks, to secure multi-hop routing in dynamic wireless sensor networks against harmful attackers exploiting the replay of routing information. TARF spotlighted on conviction worthiness and energy efficiency. With the idea of trust management, this model facilitates a node to keep track of the trustworthiness of its neighbors and thus to select a consistent route. With the idea of the energy watcher, our model calculates the total energy cost which is consumed by the packet to reach its destination. Trust manager has initiated the model of using two routing tables i.e default routing table and running routing table, with this concept its has become easy to find the attacker in the path. For provided that security in this paper we have used the new encryption technique i.e. Elliptic Curve Cryptography algorithm.

REFERENCES

[1] Guoxing Zhan, Weisong Shi, "Design and Implementation of TARF: Trust-Aware Routing Framework for WSNs" IEEE Transactions On Dependable And Secure Computing, vol. 9, no. 2, March/April 2012

[2] G. Zhan, W. Shi, and J. Deng, "Tarf: A Trust-Aware Routing Framework for Wireless Sensor Networks," Proc. Seventh European Conf. Wireless Sensor Networks (EWSN '10), 2010.

[3] F. Zhao and L. Guibas, Wireless Sensor Networks: An Information Processing Approach. Morgan Kaufmann, 2004.

[4] A. Wood and J. Stankovic, "Denial of Service in Sensor Networks," Computer, vol. 35, no. 10, pp. 54-62, Oct. 2002.

[5] 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, 2003.

[6] M. Jain and H. Kandwal, "A Survey on Complex Wormhole Attack in Wireless Ad Hoc Networks," Proc. Int'l Conf. Advances in Computing, Control, and Telecomm. Technologies (ACT '09), pp. 555-558, 2009.

[7] I. Krontiris, T. Giannetsos, and T. Dimitriou, "Launching a Sinkhole Attack in Wireless Sensor Networks; The Intruder Side," Proc. IEEE Int'l Conf. Wireless and Mobile Computing, Networking and Comm. (WIMOB '08), pp. 526-531, 2008.

[8] J. Newsome, E. Shi, D. Song, and A. Perrig, "The Sybil Attack in Sensor Networks: Analysis and Defenses," Proc. Third Int'l Conf. Information Processing in Sensor Networks (IPSN '04), Apr. 2004.

[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] S. Ganeriwal, L. Balzano, and M. Srivastava, "Reputation-based framework for high integrity sensor networks," ACM Trans. Sen.Netw., 2008.

[11] G. Zhan, W. Shi, and J. Deng, "Design, implementation and evaluation of tarf: A trust-aware routing framework for dynamic wsns," http://mine.cs.wayne.edu/_guoxing/tarf.pdf, Wayne State University, Tech. Rep. MIST-TR-2010-003, Oct. 2010.

[12] Daniela Tulone, "A Resource–efficient Time Estimation for Wireless Sensor Networks," Department of Computer Science University of Pisa