A Framework for Trust-based Cluster Head Election in Wireless Sensor Networks

Garth V. Crosby, Niki Pissinou, James Gadze
Florida International University
Telecommunication and Information Technology Institute
Miami, Florida
[gcros001, niki.pissinou,jgadz001]@fiu.edu

Abstract

The election of a malicious or compromised node as the cluster head is one of the most significant breaches in cluster-based wireless sensor networks. In light of this, we are introducing a distributed trust-based framework and a mechanism for the election of trustworthy cluster heads. Our proposed mechanism reduces the likelihood of compromised and or malicious nodes from being selected as cluster heads. Our premise is that while individual nodes may still be prone to attack, a significant vulnerability is addressed if we prevent the election of compromised cluster heads. We do not seek a 'cure for all solution' rather we introduce a framework and a mechanism to address a potentially significant security breach. We performed an evaluation of our approach and the power consumption of our model, by simulations. The results indicate clear advantages of our approach in preventing the election of untrustworthy cluster heads.

1. Introduction*

Security and trust are two related and inseparable concepts. We cannot experience security without prior assumptions of trusts and, establishing metrics of trust must be done in a secure environment. For example, secrecy is usually achieved by encrypting communication. In this instance, an encryption key is shared among authorized nodes. It is assumed that adversarial nodes are prevented from decrypting the messages because they are not in possession of the encryption/decryption key. Thus, the communication between nodes is secure (secret).

However, the communication will only be secret if the initial assumption of trust is true. What if adversarial nodes had the keys from the beginning of the network? Therefore, it is made clear that in order to ensure security it is necessary that the encryption/decryption keys be distributed to only trusted nodes. Without this premise of trust reasonable levels of security cannot be achieved.

Wireless sensor nodes are vulnerable to physical compromise. Tamper proof techniques are not cost effective for non-mission critical commodity wireless sensor network [1]. This makes them particularly prone to security breaches via the physical extraction of cryptographic material. Wireless sensor networks pose unique new challenges which prevent direct application of traditional security techniques[2]. For economic viability, sensors nodes are limited in power (which usually cannot be replenished), computation capabilities, bandwidth and memory. The limitation of memory and processing capability makes public key cryptography and digital signature infeasible. In addition, the limited power of these tiny senor nodes makes the communication overhead of traditional security algorithm unbearable. Cryptographic techniques such as symmetric encryption are particularly useful in wireless sensor networks which are inherently susceptible to eavesdropping. Cryptographic techniques however, do not offer sufficient protection to the network in the case of compromised nodes. This is because compromised nodes are already participating in the network and thus would have already had all the required cryptographic material. This makes it necessary to develop a trust framework so that

* This work was supported in part by grants from the U.S. Dept. of Defense, U.S. Dept. of Transportation and the National Science Foundation.

Proceedings of the Second IEEE Workshop on Dependability and Security in Sensor Networks and Systems (DSSNS'06)
0-7695-2529-6/06 $20.00 © 2006 IEEE

wireless sensor network can function effectively even in an environment where nodes are compromised. It then becomes essential to focus on the cluster heads since they are above average in their importance to the proper functioning of the network.

Clustering provides one of the best solutions for communication in sensor networks due to its inherent energy saving qualities and its suitability for highly scalable networks. Clustering naturally facilitates data aggregation, an energy efficient technique where nodes forwards to a cluster head for processing and fusion before transmitting to base station. Clustering can be extremely effective in multicast, anycast, or broadcast communication. However, to the best of our knowledge, all of the cluster based protocol and cluster formation algorithm that have been proposed assume that the wireless senor nodes are trustworthy [3, 4]. This assumption may naturally lead to the selection (or election) of a compromised or malicious node to be the cluster head. Having a malicious cluster-head severely compromises the security and usability of the network.

It has been demonstrated [5] that if 5% of the nodes misbehave then more than 60% of the routes in a grid sensor network and more than 35% of the routes in a randomly placed sensor network, would be infected. For 10% of misbehaving nodes the figures are 88% and 54% respectively [5]. These results imply that in a cluster-based protocol such as LEACH in which optimally 5% of the nodes are cluster heads[3], it is likely that a significant portion of the network can be paralyzed or the entire network disabled, in the worst case scenario, if these cluster heads are compromised.

In this paper, we present a framework for distributed trust in wireless sensor networks, a trust model with a novel quantitative measure of trust and, a mechanism that elects trustworthy cluster heads. The remainder of this paper is organized as follows. In section 2, we present related works. In section 3, we describe our distributive trust framework. In section 4 we describe our cluster head election mechanism. We present a high level description of our modeling of trust in section 5. In section 6, we present our simulations and analyses. We conclude in section 7.

2. Related Work

In recent time the issue of security in wireless sensor networks has been addressed in [2, 6-11] and [13-14]. In [6] the authors presented a comprehensive assessment of various denial-of-service (DoS) attacks and counter measures and how these apply to wireless sensor networks. These attacks are presented based on the security vulnerability of the physical, data link, network and transport layers. In [12] the authors evaluated a number of wireless sensor network routing protocols and highlighted their weaknesses. They showed the security threats and proposed countermeasures.

Many clustering algorithms have been proposed [13-23] for ad hoc networks. Most of these are based on heuristics and attempt to generate the minimum number of clusters such that a node in any cluster is at a limited number of hops away. Some cluster based protocols attempts to create energy efficient routing scheme for sensor networks. Among these are the Low-Energy Adaptive Clustering Hierarchy protocol (LEACH) [3], Threshold sensitive Energy Efficient sensor Network protocol (TEEN) [4] and Adaptive Threshold sensitive Energy Efficient sensor Network protocol (APTEEN) [24]. These protocols are among the most promising and popular cluster based routing protocol for large-scale sensor deployment due primarily to their incorporation of data aggregation, which significantly reduces power consumption. However, these protocols rely on algorithms that assume that all nodes are trusted. The cluster heads are in most cases self elected. Little or no consideration is given to the likely scenario where compromised nodes elect themselves as cluster heads.

A number of trust based protocols for mobile ad hoc networks (MANETs) and wireless sensor networks have been proposed. We will first discuss ad hoc networks then highlight some works that specifically address wireless sensor networks. In [25], the authors proposed a secure routing solution to find an end to end route free of malicious nodes with the collaborative effort from the neighbors. Their solution also secures the network against colluding malicious nodes. A framework for computing and distributing trusts in mobile ad hoc networks is also proposed. The propose protocol is an extension of the Ad Hoc On-Demand Distance-Vector Protocol (AODV) and the authors' previous work, Trusted-AODV (T- AODV) [26, 27]. In [28], the authors proposed a technique called Security-Aware ad hoc Routing (SAR) that incorporates security attributes as parameters into ad hoc route

discovery. The protocol is an augmentation of AODV which incorporates trust level metric in the Route Request (RREQ) message. A hierarchical trust level is implemented among the nodes.

In [29], the authors proposed a framework for the establishment and management of trust in an ad hoc network without the aid of a central authority. A model is proposed in which trust is derived by assigning weights to various observable and measurable network activities or 'trust categories'. Examples of these include data packets received, control packets received, and data forwarded. The authors proposed an augmentation of the Dynamic Source Routing (DSR) protocol that uses passive acknowledgements with nodes operating in promiscuous mode to observe network communication. Each node is able to compute the trust level of other nodes. Assignment of trust levels to each node facilitates a trustworthy source to destination path. While the authors did not address wireless sensor networks specifically, we found this work to be helpful in the formulation of our cluster-based distributive trust framework for wireless sensor networks.

A secure public key authentication service based on trust to prevent nodes from receiving false public keys from malicious nodes is proposed in [30]. The system does not rely on any trusted-third party. The trust model follows the "web of trust" approach. The model uses digital signature as its form of introduction. Any node signs another's public key with its own private key to establish a web or trust. The nodes in the network monitor each other and update their trust table, which is stored at each node, accordingly. The public key management mechanism endures the false certificate issued by dishonest users and malicious nodes, and avoids them to be selected as introducing nodes. The use of digital signature makes this approach impractical for sensor network due to the limited power and computational capacity of wireless sensor nodes.

All of the trust based papers discussed so far were developed for ad hoc networks and were not necessarily suitable for wireless sensor network due to the power, memory and computational requirement of the nodes. We now discuss proposals that were specifically designed for wireless sensor networks.

A reputation-based frame work for wireless sensor network that utilizes Bayesian formulation and beta distribution is proposed in [31]. Watchdog mechanism resides in the middleware of each node and collects observable information. Second hand information is also included in the statistical computation of reputation. This information is gathered from nodes in the neighborhood. Direct observation and second hand information together facilitates a decentralized reputation based systems. The inclusion of second hand information would normally imply that the protocol is susceptible to badmouthing attack (false reporting of observed behavior). However, the authors remove this attack by allowing the nodes to only propagate good reputation information about other nodes. As the authors themselves point out, this resiliency comes at the cost of system efficiency as now the nodes cannot exchange their bad experiences about malicious/faulty nodes in the network. In our trust model we reduce the likelihood of badmouth attack by allowing nodes to share trust information only with the cluster head. Each node maintains independent trust tables based on direct observation. Any false information good or bad is weighed against information obtained from the other nodes via the election process. This approach efficiently deals with the threat of badmouthing attack without any loss of system efficiency.

In [1], the authors proposed a 'key infection' protocol that establishes a trust framework for the distribution of keys in a non-critical commodity sensor network. The nodes broadcast keying material as they are deployed and begin making contact with other nodes. Nodes gradually increase their broadcast transmission power until contact is made with another node. On contact, keying materials are exchanged in plain text with each other. One of the premises of this paper is that in a non-mission critical commodity wireless sensor network it is reasonable to assume that adversarial nodes are not present at the set up phase of the network. The authors argue that economic factors would prevent adversarial nodes at setup since this would require the adversary to place many nodes in various locations with the hope that a network will be deployed in one of those locations. This would be a highly costly approach for the adversary and is considered impractical for non-mission critical commodity wireless sensor networks. We agree with the authors and incorporate their 'real world' attack model that excludes the existence of a global adversary during the network setup phase in our framework.

3. Trust-based Frame-work

Our primary goal is to develop a trust-based framework for cluster-based wireless sensor networks and, a mechanism that reduces the likelihood of compromised or malicious nodes being selected (or elected) as cluster heads. We make a number of assumptions. Firstly, a reliable link layer protocol and cluster formation algorithm is assumed. Once the clusters are formed they maintain the same members, except for cases where nodes are blacklisted, die or when new nodes join the network (see Fig. 1). All the nodes communicate via a shared bidirectional wireless channel and operate in the promiscuous mode, that is, if node A sends a message to node C via node B, then node A can hear if node B forwarded that message unto node C, the destination. We do not consider key distribution but we assume that each node has three keys; a master, cluster and pairwise. The master key is shared by every node and facilitate broadcast by the base station. Members of each cluster share the cluster key. Each cluster has a different cluster key. This key facilitates multicasting communication from the base station to a cluster and also group communication within the clusters themselves. The pairwise key allows node-to-node communication.

We assume synchronization and a time division multiplexing (TDM) scheduling for communication within a cluster. That is, each node within the cluster is given a time frame in which to communicate. This prevents collision of messages among cluster members and facilitates sleep-wake schedules. We recognized that using TDM inherently limits the number of nodes in a cluster. However, we relax that constraint in our mechanism. Finally, we assume that the nodes have unique local IDs. This assumption of unique local IDs should not be confused with the global ID, analogous to IP address, which many believe is infeasible for wireless sensor nodes.

We have considered a motivated attacker that attempts to become a cluster head via malicious or compromised nodes after the setup phase of the network. We envision that non-critical commodity wireless sensor nodes (non-military and non-mission critical applications) will be cheap, under a dollar per node.

As such, it would not be cost effective to implement tamper proof techniques in these nodes. As a result of this, it would be quite possible for a motivated attacker to recover valuable cryptographic information through

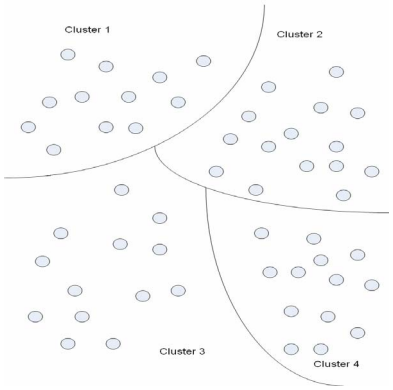

Figure 1. Clusters in wireless sensor network

physical extraction and then redeploy these nodes in the network. Our trust based mechanism aims at primarily preventing adversarial or compromised nodes from becoming cluster heads.

3.1. Architecture

After setup, the cluster heads create a time division multiplexing (TDM) schedule and inform each cluster member. The nodes are actively transmitting or listening for a period of the time and off the remainder. The nodes transmit only at their scheduled time. This allows the nodes to listen to the communication in their respective clusters. It is through this passive listening that the nodes are able to develop trust relationships with their neighbors. Nodes that constantly drop packets or which behave in a selective or selfish manner can be easily detected by their neighbors. Each node stores and maintains a trust table of its neighbors. The details of this table are discussed later in section 5.

To make clear our architecture we provide the following example. Consider the case where node A is transmitting sensed data to the cluster head as shown in Figure 2.

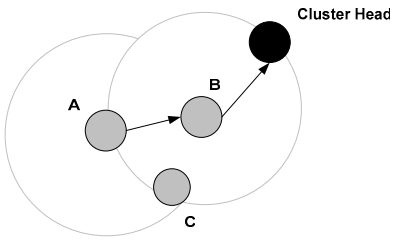

Figure 2. Communication from node to cluster head

While node A is transmitting, all the other nodes in the cluster are listening. Every node can hear the transmission of all the other nodes in their broadcast range which we define here as their neighborhood. The neighborhood (broadcast range) of node A and node B is depicted by the large circles surrounding the respective nodes. Here we see that node C is in the neighborhood of both Node A and Node B. Therefore, node C would be able to observe all the transmissions of node A and B. By simply observing the transmission of the message from node A to the cluster head via node B, node C is able to establish trust parameters for node B through computations that is discussed in section 5. This would be a measure of the reliability of node B in forwarding the message from node A to its destination, the cluster head. In addition, since node A is operating in promiscuous mode it would also be able to establish its own trust parameter in respect to node B, based on direct observation. Likewise, each node is able to observe and record the behavior of all the other nodes in its neighborhood. This leads to the maintenance of local independent trust tables at each node.

4. Cluster Head Election Mechanism

In our scheme the cluster head performs the usual functions such as data aggregation, fusion and higher level transmission to the base station. At the outset, cluster heads are self-elected [3]. We allow the self-election for the first sets of cluster-heads. This is consistent with our initial assumption that there are no adversarial nodes at setup.

When the clusters are established the cluster head schedules the transmission of each member in a TDM manner and inform all the members. When the current cluster head's battery power level falls below a predetermined threshold or serve for a predetermined period of time, it broadcasts (within the cluster) a *new election* message. All the nodes then vote for a new cluster head by using secret ballot. This is done by replying to the *new election* message with its choice of candidate. The reply, or vote, is encrypted with the pairwise key with the cluster head. Neighbors therefore have no idea of the political affiliation of each other since the key is private and, different for each node–cluster head pair. The top pick from its list of trusted neighbors is selected as the node's candidate. The current cluster head then tallies the votes and decides the winner based on simple majority. The node with the second highest number of votes is selected as the vice cluster head. The purpose of the vice cluster head is to assume cluster head function in the event that the newly elected cluster head fails before handing over to its successor. At the completion of tallying, the cluster head multicast the winner and runner-up to all the members of the cluster.

For greater integrity the new winner and runner-up have to pass a challenge-response from the cluster head before they are allowed to take up office. If one or both of them fail the incumbent cluster head informs the cluster members and, initiate a new election for the replacement of the corrupt node(s), which we define here as the nodes that did not pass the challenge-response. The corrupt node(s) are blacklisted in the cluster nodes' trust tables by setting its trust level value to -1. Once a node is set to -1 no further trust level update is done.

```
 if power_level( ) <= threshold or clusterhead_duration
>= predetermined_time
{
New_Election( ) {
      broadcast new_election( )
      count nominees( ) //tally the votes for each
nominee
   if Tie
      top_nominee = randomly_select_nominee( )
   else
      top_nominee= max_count( )
   end if
   //sends challenge response to top_nominee
   if challenge_response( ) =pass
      new_head = top_nominee
      broadcast new_head
   else
      blacklisted=top_nominee
      broadcast blacklisted
      New_Election( )
   end if
end} // end of function New_Election
}
```

Figure 3. Cluster head election procedure

Occasionally the cluster head will broadcast a *not trusted* message. In this case, nodes select the least trusted neighbor and reply to the cluster head in a similar manner to the voting process. The cluster head tallies the *no trust* messages and selects the node that is least trusted by the most nodes. That node is then given a challenge-response by the cluster head. If it fails, it is blacklisted. If it passes, the cluster members are informed as such. However, they are not obliged to improve the trust level of the node in question

because it may not be corrupted but may still be unreliable and as such deserves a low trust level.

The procedure in Figure 3 gives a high level description of the action of the current cluster head in the election of a new cluster head. A similar procedure applies when electing the vice cluster head.

We also address trust levels storage and distribution. Trust modeling refers to a technique for the digital computation of trust. *Trust* can be defined generally as the expectation of one person about the actions of others. It is used by the first person to make a choice, when an action must be taken before the actions of others are known [31] . With respect to wireless sensor networks we can define *trust level* as a measure of the predictability of a node to reliably relay messages based on direct or indirect observation of past behavior. Here indirect observations refer to information a node obtained from trusted nodes.

5.1 Trust Parameters

The trust parameters are observable and measurable network events. Each node has a watchdog mechanism that allows it to monitor the network events of other nodes. Using the information obtained through monitoring enable the nodes to compute and store trust levels for its neighbors.

A node can get information about the successful transmission of any packet that it sent, via passive acknowledgement. In passive acknowledgement the sender node places itself in promiscuous mode after the transmission of any packet so as to overhear the retransmission by the recipient node. In addition, while one node is transmitting, all other nodes in the neighborhood are listening so that they can also determine if the message was successfully delivered. In cases where the messages are to be forwarded, neighborhood nodes can tell if the message was modified before retransmission by comparing with the message in its buffer. This requires the nodes to store the messages from their neighbors for at least one TDMA frame. Generally, passive acknowledgement provides us with the following information [29]:

1. Data packets are dumped and not re-transmitted
2. Data contents have been fallaciously modified
3. Unique addresses have been spoofed

This method of passive acknowledgement can also apply to control packets. The number of these acknowledgements occurring with respect to every node is maintained and tabulated as shown in Table 1.

Table 1. Trust table based on passive acknowledgements for cluster with n nodes

Nodes	X_o	….	X_n
Data Packet Received for forward, RF_N	.		.
Data Packet Forwarded, F_N	.		.
Data Packet Modified, DM_N	.		.
Data Packet Address Modified, AM_N	.		.
Control Packet Received for forward, CRF_N	.		.
Control Packet Forwarded, CF_N	.		.
Control Packet Modified, CM_N	.		.
Control Packet Address Modified, CAM_N	.		.
Trust Level, $T_N(X_i)$	.		.

5.2 Computation of Trust Level

A trust level, denoted by $T_N(X_i)$ about X_i, where $0 \le i \le n$, is created at each node N. This is the trust level node N has computed and assigned to node X_i based on observation of node X_i's past behavior. The $T_N(X_i)$ is computed as follows:

$$T_N(X_i) = \omega_1 d_1 + \omega_2 d_2 + \omega_3 d_3 + \omega_4 c_1 + \omega_5 c_2 + \omega_6 c_3 + \gamma$$

where ω_1 to ω_6 are weights and γ is a predetermined constant that is set to equal to the

average packet drop rate of the network; d_1, d_2, d_3 and c_1, c_2, c_3 are related to the data packets and control packets respectively. They are computed as follows (the notations are specified in Table 1):

$$d_1 = \frac{F_N(X_i)}{RF_N(X_i)} \leq 1$$

$$d_2 = 1 - {DM_N(X_i)}/{F_N(X_i)}$$

$$d_3 = 1 - {AM_N(X_i)}/{F_N(X_i)}$$

$$c_1 = \frac{CF_N(X_i)}{CRF_N(X_i)} \leq 1$$

$$c_2 = 1 - {CM_N(X_i)}/{CF_N(X_i)}$$

$$c_3 = 1 - {CAM_N(X_i)}/{CF_N(X_i)}$$

5.2 Trust level Storage and Distribution

Each node stores a trust table in which it records the trust levels of each of its neighbors. Neighbors are confined to those within the broadcast radius of the node. The mechanism does not encourage sharing of trust information among neighbors and the node does not record a trust level for itself. Trust levels are only sent to the cluster head upon request. This mechanism reduces the effect of bad mouthing, since trust computation is not based on second hand observation; except by the cluster head in the final tally, when all the votes are counted. Also, since the nodes do not record their own trust level it is less likely for malicious nodes to upgrade themselves to high trust levels.

6. Simulation

In this section, we use simulation to study the performance of our model. We use OPNET [32] as our main simulation platform. First, we evaluated the energy efficiency and power consumption requirement of our model. We then assessed the capability of our model in preventing compromised nodes from being selected as the cluster head.

6.1 Environment Setup

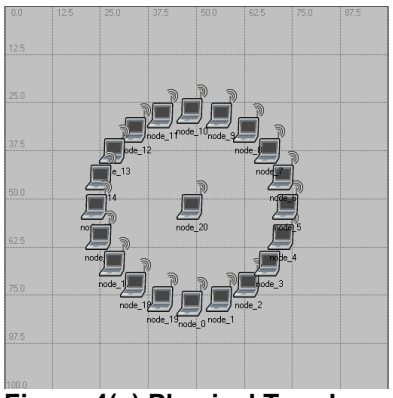

Figure 4(a) Physical Topology of Cluster

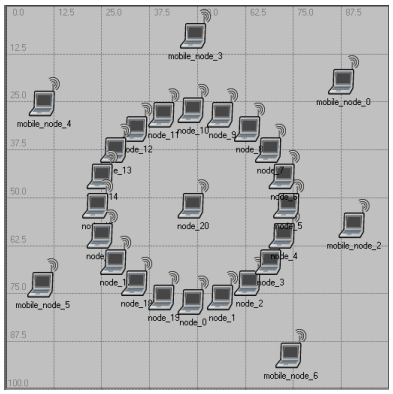

Figure 4(b) Physical topology used in second setup

We have two different environment setups. We use the first to evaluate the power consumption of the nodes and, the second to evaluate the capability of our mechanism in preventing compromised nodes from being selected as the cluster head.

In the first setup, a 20 node cluster arranged as shown in Figure 4(a) is placed on a 100m^2 area. The diameter of the circular arrangement is 50 meters. The cluster head is placed in the center as shown. The nodes transmission distance is 50 meters. A free space propagation model is assumed with a data rate set a 2Mb/s. Packet lengths are 10kbit for data packets. The data packets are generated every one second [33]. The data packet format is shown in Figure 5. We use a simple TDMA based MAC with only data packets and two types of control packets. We use 2 byte control packets for synchronization and 1byte

control packets as 'new election messages'. These control messages are sent periodically by the cluster head to all the nodes in the cluster.

Source ID	Dest. ID	Vote	Payload

Figure 5. Data Packet Fields

The nodes transmit only to the cluster at their schedule time. However, all nodes are able to hear the transmissions. On reception of the new election message the nodes include the node ID of their most trusted neighbor in the vote field of the packet that is next in line for transmission.

For the second setup we use a 20 node cluster with dimension similar to the previous setup. In addition, we include additional nodes presumably from other nearby clusters as shown in Figure 4(b). These nodes transmit at 10kbps to a random subset of nodes in the cluster, which are within their transmission range. The transmission range for these nodes is 20 meters. These additional nodes are for the purposes of relaying data from nearby clusters. We interpret all transmission of these nodes as 'data received for forward'. We simplify the simulation study by observing only 'data received for forward' and 'data forwarded by the node'. Address modification and observation of control messages are not considered. We believe that while it would be interesting to examine these as well, the focus on data packets is sufficient to validate the usefulness of our proposal.

The cluster head runs our cluster election algorithm. We omit the challenge response procedure, assuming that once selected the new cluster head has the necessary cryptographic material. This narrows our study to compromised nodes as oppose to compromised and malicious nodes.

We were interested in testing the capability of our algorithm in discerning between trusted and untrustworthy nodes. Therefore, compromised nodes were systematically introduced in the setup by setting the node's packet drop rate to 40%. The packet drop probabilities of the other nodes were set to 0.01 [33]. This was done to make the simulation more realistic. The compromise nodes ignore the prescribed selection routine and randomly votes for nodes. This was implemented since by intuition we do not expect compromised nodes to report truthfully. In the next section, we present results that show the capability of the algorithm in preventing the selection of compromised nodes as cluster heads.

6.2 Analysis of Results

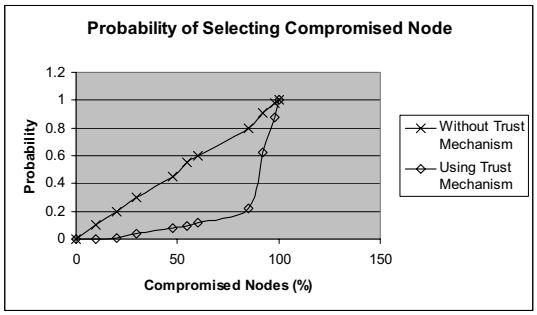

Figure 6. Probability of Selecting Compromised Node as CH

Figure 6 shows the advantage of our selection mechanism over a similar cluster that doesn't employ our trust-based election mechanism. For clusters with less than 15% of compromised nodes our mechanism almost never selects a compromised node. This demonstrates the effectiveness of our mechanism in securing cluster based wireless sensor networks. There is an expected linear increase over time, however, the probability increase rapidly after 85% of the nodes were compromised. This can be explain by an accumulation of errors at the node that makes it increasingly difficult to discern between compromised nodes and uncompromised node in light of the packet drop rate and the false voting of compromised nodes.

While we did not fully investigate the effects of collusion we suggest that effective collusion would require more than 10 % of the nodes to be compromised. This is based on some preliminary simulation results.

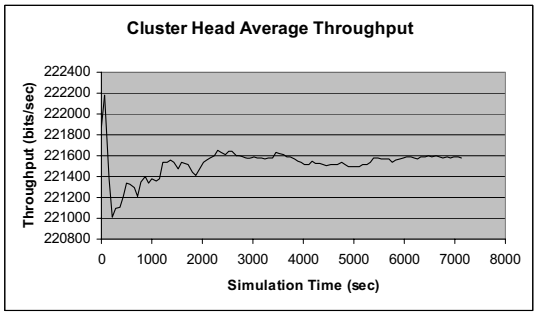

Figure 7. Average Cluster Head Throughput

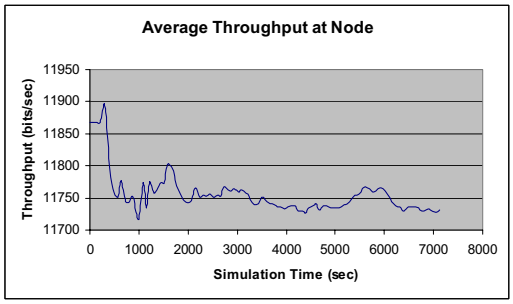

Figure 8. Average Node Throughput

Figures 7 and 8 show the average throughputs of the cluster-head node and a regular node. The initial peak in the graphs can be explained by the fact that the cluster nodes were scheduled to start transmission of data closer in time with respect to each other, than the inter-arrival time of the packets. Afterwards the throughput was maintained at approximately 11,750 bits/sec. Based on these results and using the communication energy model in [3] we can obtain some estimate for the power consumption of our model. As an example, if a 1-volt AAA battery with 750mWh is used for each node, the battery can last for 18 days assuming that the node serves a short period as a cluster head. This is a fairly good lifetime for the node given that we have employed a simple MAC, without any energy optimization algorithm.

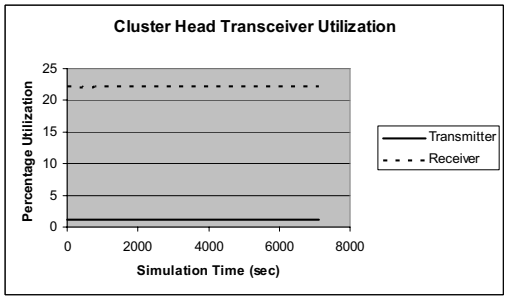

Figure 9. Cluster Head Transceiver Utilization

In Figure 9, we compare the utilization of the CH transmitter and receiver. We see that the receiver is on 22% of the time while the transmitter is on about 1% of the time. This means that the transceiver circuitry can be off for approximately 77% of the time, since there is no need for listening outside of the scheduled time for reception. Here we get a duty cycle of 23% for the transceiver circuitry. However, we note that this figure is dependent upon the number of nodes in the cluster, the channel data rate and the data rate of the nodes.

7. Conclusion

This paper describes a trust based framework and a mechanism for the election of trustworthy cluster heads. To this end, we have proposed a trust model with a novel quantitative measure of trust. Our framework is design in the context of a cluster based network model with nodes that have unique local IDs. We assess our model based on power consumption and its ability to prevent compromised nodes from becoming cluster heads. Our approach decreases the likelihood of malicious or compromised nodes from becoming cluster heads.

Our trust model is most suitable for wireless sensor networks due to its minimal energy and computational requirement. We intend to examine the scalability of our model through comprehensive simulations. We also intend to test the validity and examine the implication of violating some of our initial assumptions. Further assessment of the capability of our trust-based mechanism against colluding nodes will be done.

References

[1] R. Anderson, H. Chan, and A. Perrig, "Key Infection: Smart Trust for Smart Dust," presented at The 12th IEEE International Conference on Network Protocols (ICNP'04), Berlin, Germany, October 2004.

[2] S. Slijepcevic, M. Potkonjak, V. Tsiatsis, S. Zimbeck, and M. B. Srivastava, "On Communication Security in Wireless Ad-Hoc Sensor Networks," presented at The Eleventh IEEE International Workshops on Enabling Technologies:Infrastructure for Collaborative Enterprises, 2002.

[3] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishan, "Energy Efficient Communication Protocol for Wireless Microsensor Networks," presented at The 33rd International Conference on System Sciences (HICSS 2000), Hawaii, 2000.

[4] A. Manjeshwar and D. P. Agrawal, "TEEN: A Protocol for Enhanced Efficiency in Wireless Sensor Networks," presented at The 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, April 2001.

[5] S. Tanachaiwiwat, P. Dave, R. Bhindwale, and A. Helmy, "Location-centric isolation of misbehavior and trust routing in energy-constrained sensor networks," presented at 2004 IEEE International

Conference on Performance, Computing, and Communications, 2004.

[6] A. D. Wood and J. A. Stankovic, "Denial Of Service in Sensor Networks," in *IEEE Computer*, vol. 35, 2002, pp. 54-62.

[7] S. Wu, Y. Pawar, and N. Aeron, "Security issues in sensor network," October 2003.

[8] K. Jones, A. Wadaa, S. Olariu, L.Wilson, and M.Eltoweissy, "Towards a New Paradigm for Securing Wireless Sensor Networks," presented at ACM New Security Paradigms Workshop, Ascona,Switzerland, 2003.

[9] A. Perrig, J. Stankovic, and D. Wagner, "Security in Wireless Sensor Networks," in *CACM*, vol. 47, 2004.

[10] E. Callaway, "Secure Low-Power Operation of Wireless Sensor Networks," in *Sensors Online*, vol. 21, 2004.

[11] C. Giovanni, "Topology for Denial of Service," Endeavour Systems Inc, White Paper July 12, 2000.

[12] C. Karlof and D. Wagner, "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures," presented at First IEEE International Workshop on Sensor Network Protocols and Applications, 2003.

[13] D. J. Baker and A.Ephremides, "The Architectural Organization of a Mobile Radio Network via a Disributed Algorithm," *IEEE Transactions on Communications*, vol. 29, pp. 1694-1701, November 1981.

[14] B. Das and V. Bharghavan, "Routing in Ad-Hoc Networks Using Minimum Connected Dominating Sets," presented at ICC, 1997.

[15] C. R. Lin and M. Gerla, "Adaptive Clustering for Mobile Wireless Networks," *Journal on Selected Areas in Communication*, vol. 15, pp. 1265-1275, September 1997.

[16] A. D. Amis, R. Prakash, T. H. P. Vuong, and D. T. Huynh, "Max-Min D-Cluster Formation in Wireless Ad Hoc Networks," presented at IEEE INFOCOM, March 2000.

[17] C. F. Chiasserini, I. Chlamtac, P. Monti, and A. Nucci, "Energy Efficient Design of Wireless Ad Hoc Networks," presented at European Wireless, February 2002.

[18] A. B. McDonald and T. Znati, "A Mobility Based Framework for Adaptive Clustering in Wireless Ad-Hoc Networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1466-1487, August 1999.

[19] M. Gerla and J. T. C. Tsai, "Multicluster, Mobile, Multimedia Radio Networks," *Wireless Networks*, vol. 1, pp. 255-265, 1995.

[20] S. Basagni, "Distributed Clustering for Ad Hoc Networks," presented at International Symposium on Parallel Architectures, Algorithms and Networks, June 1999.

[21] S. Basagni, "Distributed and Mobility-Adaptive Clustering for Multimedia Support in Multi-Hop Wireless Networks," presented at Vehicular Technology Conference, 1999.

[22] M. Chatterjee, S. K. Das, and D. Turgut, "WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks," *Journal of Cluster Computing, Special issue on Mobile Ad Hoc Networking*, pp. 193-204, 2002.

[23] A. D. Amis and R. Prakash, "Load-Balancing Clusters in Wireless Ad Hoc Networks," presented at ASSET 2000, Richardson, Texas, March 2000.

[24] A. Manjeshwar and D. P. Agrawal, "APTEEN: A Hybrid Protocol for Efficient Routing and Comprehensive Information Retrieval in Wireless Sensor Networks," presented at the 2nd International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile computing, Fort Lauderdale, FL, April 2002.

[25] T. Ghosh, N. Pissinou, and K. Makki, "Towards Designing a Trusted Routing Solution in Mobile Ad Hoc Networks," *ACM Journal, Mobile Networks and Applications (MONET)*, Special issue on Non-Cooperative Wireless Networking and Computing,2005.

[26] N. Pissinou, T. Ghosh, and K. Makki, "Collaborative Trust-based Secure Routing in Multihop Ad Hoc Networks," presented at The Third IFIP-TC6 Networking Conference (Networking '04), Athens, Greece, 2004.

[27] T. Ghosh, N. Pissinou, and K. Makki, "Collaborative Trust-based Secure Routing Against Colluding Malicious Nodes in Multi-hop Ad Hoc Networks," presented at Annual Conference on Local Computer Networks (LCN), Tampa, USA, 2004.

[28] S. Yi, P. Naldurg, and R. Kravets, "Security-Aware Ad hoc Routing for Wireless Networks," UIUCDCS-R-2001-2241, August 2001.

[29] A. A. Pirzada and C. McDonald, "Establishing Trust in Pure Ad-hoc Networks," presented at The 27th Australasian Computer Science Conference, Dunedin, New Zealand, 2004.

[30] E. C. H. Ngai and M. R. Lyu, "Trust and Clustering Based Authentication Services in Mobile Ad Hoc Networks," presented at The 24th International Conference on Distributed Computing Systems Workshops (ICDCSW'04), Tokyo, Japan, March 2004.

[31] S. Ganeriwal and M. B. Srivastava, "Reputation-based Framework for High Integrity Sensor Networks," presented at ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '04), Washington, D.C., USA, October 25, 2004.

[32] www.opnet.com.

[33] K. Arisha, M. Youssef, and M. Younis, "Energy-Aware TDMA-Based MAC for Sensor Networks," presented at The IEEE Integrated Management of Power Aware Communications, Computing and Networking (IMPACT'02), New York City, New York, May 2002.