A Novel CDS-based Reputation Monitoring System for Wireless Sensor Networks

Avinash Srinivasan, Feng Li, and Jie Wu *
Department of Computer Science and Engineering
Florida Atlantic University
Boca Raton, FL 33431
Email: {asriniva@, fli4@, jie@cse.}fau.edu

Abstract

Reputation and Trust-based Monitoring Systems (RTMSs) have provided a ubiquitous framework for secure Wireless Sensor Network (WSN) computing. Employing sensors for neighborhood monitoring, which is secondary to their intended duties, depletes valuable and scarce resources which is counter-productive in WSNs. In this paper, we propose a novel, Connected Dominating Set (CDS)-based reputation monitoring system. Our model is the first attempt to employ a CDS-based monitoring backbone to securely aggregate the reputation of sensors without subjecting them to energy depletion or reputation pollution attacks encountered in existing reputation monitoring systems. Secure and certificateless node mobility and robustness to node replication and ID spoofing attacks are two vital by-products of our model. We confirm the performance of our model via simulation studies.

1 Introduction

Reputation and Trust-based Monitoring Systems (RTMSs) have provided a ubiquitous framework for secure Wireless Sensor Network (WSN) computing by capitalizing on the openness of the transmission medium. Sensors are highly energy-constrained and their autonomous operation in hostile, unattended territories renders them vulnerable to physical capture. Consequently, cryptography alone cannot ensure security in WSNs, since the adversary can extract all the information stored onboard the captured node, including the cryptographic keys. This scenario is commonly referred to as *insider attacks*, in which the adversary is a legitimate member of the network, and currently, [5] and [7] counter the insider attacks in WSNs.

Given that nodes build their reputation from scratch after deployment, it can take a considerable amount of time before the system is bootstrapped to the operational threshold. This can be detrimental in WSNs, since sensors are highly energy-constrained. In particular, until the RTMS can take over as the primary system, a back-up mechanism has to be in place for the system to function and to generate network activity that enables the monitoring system to build reputation for all the nodes. Therefore, using the same sensors for monitoring as well as intended network services is counter-intuitive. We believe resource-constrained sensors should be used only for required services such that the network lifetime can be prolonged. Therefore, using a set of nodes exclusively for neighborhood monitoring will be very productive and can greatly enhance network lifetime.

Building reputation based solely on direct observation can be very time-consuming. Therefore, information sharing is vital for faster convergence of the system. Information sharing is also very useful in having a more consistent local view. However, information sharing can be fatal to the system if nodes resort to a *tit-for-tat* attitude, which pollutes the reputation values. A tit-for-tat attitude is one in which a node u, on receiving a low rating from node v, decrements its rating of v as a retaliation.

In light of the above discussion, we propose a novel, CDS-based monitoring system to function as the monitoring backbone, thereby discharging sensors from their neighborhood monitoring obligations. Our model also prevents nodes from polluting reputation. The monitoring backbone is formed by a set of special nodes referred to as monitor nodes, which we will discuss further in Section **??**. In our model, the sensors maintain reputation values for all the nodes in their neighborhood, which are provided to them by their manager, i.e., the monitor node they belong to. If a node belongs to the jurisdiction of more than one monitor

*This work was supported in part by NSF grants ANI 0073736, EIA 0130806, CCR 0329741, CNS 0422762, CNS 0434533, CNS 0531410, and CNS 0626240.

node, then the monitor node with the highest priority will be its manager. When a monitor node goes to sleep, coverage of the network is affected, which in turn jeopardizes the security of the entire system. This issue can be addressed by adding new monitor nodes to the CDS to collectively cover the jurisdiction of the sleeping node. The CDS-Monitor node intending to sleep sends out a copy of its gathered information to all the neighboring CDS-Monitor nodes. Now, when new monitor nodes are added to the CDS, they will have at least one CDS-Monitor neighbor of the sleeping node as their neighbor from whom they get the information of the sleeping node. This enables the newly added CDS-Monitor node(s) to continue monitoring the neighborhood from the point where the sleeping node left off. However, the challenge in our model lies in maintaining and reconstructing the CDS when one or more nodes go to sleep. Also, deciding on how much information to share with each neighboring monitor node prior to sleeping, to keep the redundancy to a bare minimum, can be challenging.

In RTMS, node mobility poses a new challenge which can be addressed by answering two fundamental questions: (1) Do nodes transfer their existing reputation from the current location to the new location that they intend to move to? and (2) How does a node securely transfer its accumulated reputation to the new location? This problem is addressed by our model in an effective way. Whenever a sensor node has to move to a new location, the new manager of the node will ask the previous manager of node about its behavior. The previous manager then provides the node's accumulated reputation to the new manager who will then disseminate it in its jurisdiction. This enables the node to move to a new location without having to transfer its existing reputation via cumbersome cryptographic certificates or build its reputation from scratch. We will discuss this in detail in Section 4.

Our contributions in this paper can be summarized as follows. A CDS-based reputation monitoring system has been proposed for the first time. The proposed model significantly reduces system convergence time. Our model mitigates the burden of computation and communication overhead on energy-constrained sensors by discharging them from reputation monitoring and processing obligations. Our scheme can thwart node replication and ID spoofing attacks very effectively. The proposed model ensures certificateless node mobility by securely and efficiently bootstrapping a mobile sensor node in its new location. The proposed model is robust to reputation pollution caused by either information asymmetry attacks or a tit-for-tat attitude of nodes. We evaluate the performance of our model through simulation and analysis.

2 Preliminaries

RTMS overview. In a RTMS, nodes monitor neighborhood behavior using a *watchdog*. The information gathered by virtue of direct observations using the watchdog is referred to as *firsthand* information, which is the most reliable piece of information. Direct observations are recorded in two parameters α and β, denoting good and bad behavior respectively. (α, β) is then converted into a reputation value using the Beta distribution function $Beta(\alpha, \beta)$ [1]. However, if nodes are allowed to build reputation only from firsthand information, it can be very time consuming. Hence, nodes are encouraged to publish their findings in their neighborhood, which is known as *secondhand* information. Nodes usually perform a deviation test before accepting secondhand information to mitigate information asymmetry attacks ([7]) causing reputation pollution. When a decision has to be made for choosing a partner for any network activity, nodes use the accumulated reputation values to choose the most trustworthy neighbor.

CDS overview. In an unweighted, undirected graph $G = (V, E)$, with V as the set of vertices and E as the set of edges, a node u dominates another node v if and only if $u = v$ or u and v are adjacent. Let the CDS of G be a set of vertices $V_{CDS} \subset V$.

Definition 1 *A connected dominating set of a graph $G = (V, E)$ is a set of vertices $V_{CDS} \subset V$ such that, for every vertex $v \in V - V_{CDS}$, there is at least one vertex $u \in V_{CDS}$ that dominates v, and V_{CDS} is connected.*

Once the CDS is obtained, Dai and Wu's *Rule-k* algorithm [4] is applied to reduce the size of the CDS. A node u from the CDS can be unmarked (pruned) if u is completely covered by a subset of its neighbors N' and the following conditions are satisfied: (1) Subgraph induced by N' is connected; (2) Each neighbor of u is adjacent to at least one node in N'; and (3) All nodes in N' have a higher priority than u.

Motivation and assumptions. The CDS of a network is a set such that every node in the network is either in the set or a neighbor of one or more nodes in the set. This is a highly desirable property for monitoring systems because, by having the CDS nodes monitor the nodes in their jurisdiction, we can monitor the entire network with very little resource expenditure. Following are the underlying assumptions of our proposal. Node failure is assumed to occur when a node goes to sleep and no new nodes are added to the network after initial deployment. Sensor nodes are mobile and monitor nodes are static with large storage and processing capacities and a watchdog for monitoring. Monitor nodes cannot be tampered with and are always trusted. We assume that a CDS of the network exists under any given scenario.

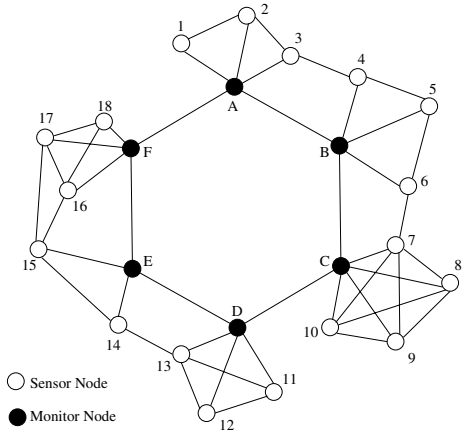


Figure 1. Schematic diagram representing our model on a network of 18 sensor nodes and 6 monitor nodes.

3 The Monitoring Backbone

There are two types of nodes in our model: sensors and monitors. Each node in the network is randomly assigned a unique ID prior to deployment. The neighborhood of a node consists of three sets of nodes: sensors, CDS-Monitors, and non-CDS monitors. In our system, the information is primarily firsthand, and secondhand information is used for bootstrapping sensors when they move to a new region, which will be discussed further in Section 4. The monitoring backbone, a CDS, is constituted by a subset of monitor nodes here after referred to as CDS-Monitors. If a node belongs to the jurisdiction of more than one CDS-Monitor, then the one with the highest priority will be its manager. For example, in Fig. 2 (b), node 2 belongs to the jurisdiction of CDS-Monitors A and B. Based on node priority $A > B$, A will be 2's manager.

When nodes go to sleep, coverage of the network is affected, which in turn jeopardizes the security of the entire system. This issue can be addressed by adding new monitor nodes to the CDS to collectively cover the jurisdiction of the CDS-Monitor intending to sleep. However, if the newly added CDS-Monitors were to start building reputation of nodes in their jurisdiction from scratch, then it could take a significant amount of time for the neighborhood to be bootstrapped. To overcome this problem, the CDS-Monitor intending to sleep sends out a copy of its gathered information to all the neighboring CDS-Monitor. This enables the newly added CDS-Monitor(s) to continue monitoring the neighborhood from where the sleeping node left.

Reputation computation. In RTMSs, where there is information sharing, nodes are vulnerable to reputation pollution either due to information asymmetry attacks or due to nodes adopting a tit-for-tat attitude. This kind of attitude among nodes leads to pollution and inconsistency in reputation values and eventually to instability of the entire system. These problems can be overcome by using the monitoring backbone proposed in this paper for both garnering observations and reputation computation.

In our model, CDS-Monitors constantly update their observation parameters α and β for each sensor node in their jurisdiction, which is then converted into a reputation value. The CDS-Monitors then disseminate the reputation values in their jurisdiction. Since a malicious sensor node cannot harm the CDS-Monitor for publishing low reputation values, there is no room for it to adopt a tit-for-tat attitude. The main advantage of this model is that there is no computation overhead imposed on sensor nodes, which preserves scarce resources. If a sensor node has a neighbor which belongs to the jurisdiction of a different CDS-Monitor, then it requests that its manager provide it with the required information. The requesting node's manager then contacts the requested node's manager to obtain this information and provide it to the requesting node. For instance, in Fig. 1, when node 4 needs the reputation of node 3, it makes a request to its manager B. B, in turn, makes a request to A, who is the manager of node 3. A then provides the computed reputation value of node 3 to B, which then provides it to node 4. Though this process involves a couple of extra hops of information transmission, note that from the sensor nodes perspective, its still a two-step process: request and response, although it has a slightly higher delay.

Our reputation computation model is simple yet robust against information asymmetry attacks. Information asymmetry attacks specific to reputation monitoring systems were introduced by Srinivasan et al in [7], in the context of beacon-based sensor localization. In the proposed model, each monitor node uses its own observation for computing the reputation of every node in its jurisdiction. Since only firsthand information is used, it prevents reputation pollution arising due to information asymmetry attacks.

Maintenance of Monitoring Backbone. Since sensor nodes are mobile, the CDS has to be updated after node movement. The CDS has to be recomputed even when CDS-Monitors go to sleep. There are two important issues that need to be addressed for CDS recomputation: (1) Are there sufficient monitor nodes to cover the entire network even if some of the current CDS nodes go to sleep? and (2) When a node goes to sleep, how does it distribute the gathered information to the neighboring CDS nodes?

To address the issue raised by the first question, we assume that a sufficiently large number of monitor nodes are uniformly distributed during the initial deployment, along with regular sensor nodes. This guarantees that we will be able to find a monitor node to replace an existing CDS-

Algorithm 1 CDS Maintenance

for each CDS-Monitor node i intending to sleep **do**
  Send copy of accumulated reputation values to neighboring CDS-Monitor nodes;
end for
for each j that is a neighbor of i or a neighbor of CDS-Monitor neighbors of i **do**
  $V_{CDS} \leftarrow j$
end for
Execute Localized *Rule-k* Algorithm on the new V_{CDS} to prune redundant monitor nodes;
Update sensor nodes of their new manager;

Monitor when it goes to sleep or when a sensor node moves to a new location. Addressing the challenge raised by the second question is much trickier. When a node goes to sleep, the number of monitor nodes that are required to cover its jurisdiction is not fixed. The best scenario is when a single monitor node is needed to cover the jurisdiction of the CDS-Monitor going to sleep.

Therefore, when a CDS-Monitor is about to sleep, it provides a copy its information to its neighboring CDS-Monitors. The reason it chooses to provide a complete copy of its information is because more than one monitor node may be required to cover its jurisdiction. Secondly, the reason it chooses to provide a copy of the information to only its neighboring CDS-Monitors is because the new CDS-Monitor has to re-establish the broken link between the sleeping node and its waking CDS-Monitor neighbors.

For illustration, in Fig. 2 (b) according to Algorithm 1, when node A intends to sleep, it sends a copy of its information to CDS-Monitors B and G. Following this, node H, a neighbor of A, is added to the set V_{CDS}. Also, node F, a neighbor of both B and G, which are CDS-Monitor neighbors of node A, and node E, which is a neighbor of F, are added to the set V_{CDS}. Following this, the *Rule-k* Algorithm is executed to prune redundant nodes from the set V_{CDS}. Consequently, node E gets pruned from the V_{CDS}.

Finally, the two newly added nodes H and F re-establish the broken links (B, A) and (A, G), connecting the two waking CDS-Monitor neighbors B and G of A with three new links (H, G), (G, F), and (F, B). From this example, it is clear as to why it gives a copy of the information to all its CDS-Monitor neighbors. For instance, in Fig. 2 (b) if node D intends to sleep and gives a copy of its information to only one of its CDS-Monitor neighbors, say C, then the newly added CDS-Monitor E would not have the information. This would necessitate E to monitor the neighborhood building reputation of nodes in its jurisdiction from scratch, which is counter-productive.

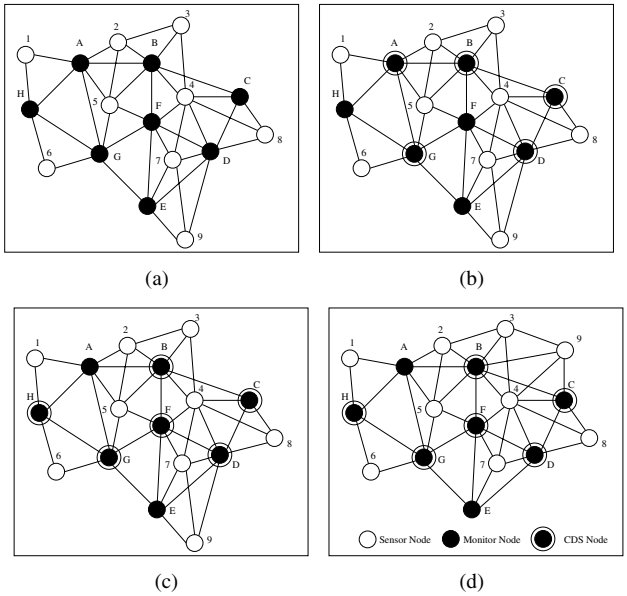


Figure 2. Mobility of nodes is confined to the square boundary. (a) Initial network (b) CDS of the network (c) Updated CDS after monitor A sleeps (d) Updated CDS after node 9 moves to a new location and joins monitor B.

4 Discussions

Secure node mobility. Currently, node mobility models under the framework of RTMSs either require the node to collect digital certificates vouching for its reputation or to build its reputation afresh in the new location. The model proposed in this paper neither requires the node to carry digital certificates nor to build reputation from scratch. In our model, node mobility can be accomplished in one of the following two ways: (1) *reactive bootstrapping*, (2) *proactive bootstrapping*. In the reactive bootstrapping mobility mode, a node simply moves to the new location and joins a new manager node and provides its previous manager's ID. The new manager then contacts the previous manager of the node to verify the node's claim of its previous location as well as collect its accumulated reputation. After authenticating the node and receiving its reputation information from the previous manager, the new manager bootstraps the node in its new location. This enables the node to start participating in network activities readily and receive cooperation from neighboring nodes.

In the proactive bootstrapping mobility model, assuming predictive mobility, a node intending to move notifies its manager of its target location. With this information, the manager can communicate with the potential new managers

in the target region and provide them with reputation information of the node. Subsequently, when the node moves into the new location, it has to merely provide its previous manager's ID to the new manager and it will be instantly bootstrapped to the reputation state that it had in the previous location. In this model we assume that there is an upper bound on time for the node to move to the new location and join the new manager. Figs. 2 (c) and (d) have captured the ideal of node mobility. In Fig. 2 (c), node 9 belongs to the jurisdiction of monitors D and E, but its manager is D (priority $D >$ priority E). Now, after moving to a new location, node 9 is in the jurisdiction of monitors B and C, and based on priority resolution, B will be its new manager.

Robustness. Our model curtails ID spoofing attacks very effectively and with little overhead. When a malicious node re-enters the network through ID spoofing, its new manager will ask it to provide its previous manager's ID. When the node fails to provide a valid manager ID, it will be immediately blacklisted and denied cooperation.

In our model, there is no way for the adversary to launch a node replication attack, since whenever a node joins a network location, it has to furnish its previous manager's ID. With node replication attacks, replicated nodes can provide their previous manager's ID. However, when the previous manager is contacted, if indeed there has been a node replication attack, then the previous manager will still claim its dominance over the node under consideration. Immediately, an alarm will be raised and the corresponding node's ID will be broadcast along the CDS backbone to all monitor nodes and all copies of the node will be blacklisted and isolated.

5 Related Work

Numerous RTMSs, such as CORE [2], RFSN [5], and DRBTS [7] have been developed to stimulate node cooperation. In most of these models, nodes build their own view based on personal observations as well as the recommendations from neighbors. Michiardi and Molva [2] proposed CORE, which has a watchdog along with a reputation mechanism to distinguish between subjective, functional, and indirect reputation, all of which are weighted to get the combined reputation of a node.

The reputation bootstrap problem has been presented as a key challenge in [3], and extended bootstrap periods for newcomer nodes are considered to be burdensome. Consequently, a distributed framework is vital for nodes to securely transfer their accumulated reputation values to the target region and mitigate the newcomer problem. In [8], it has been confirmed that mobility of nodes helps in mitigating uncertainty in RTSs.

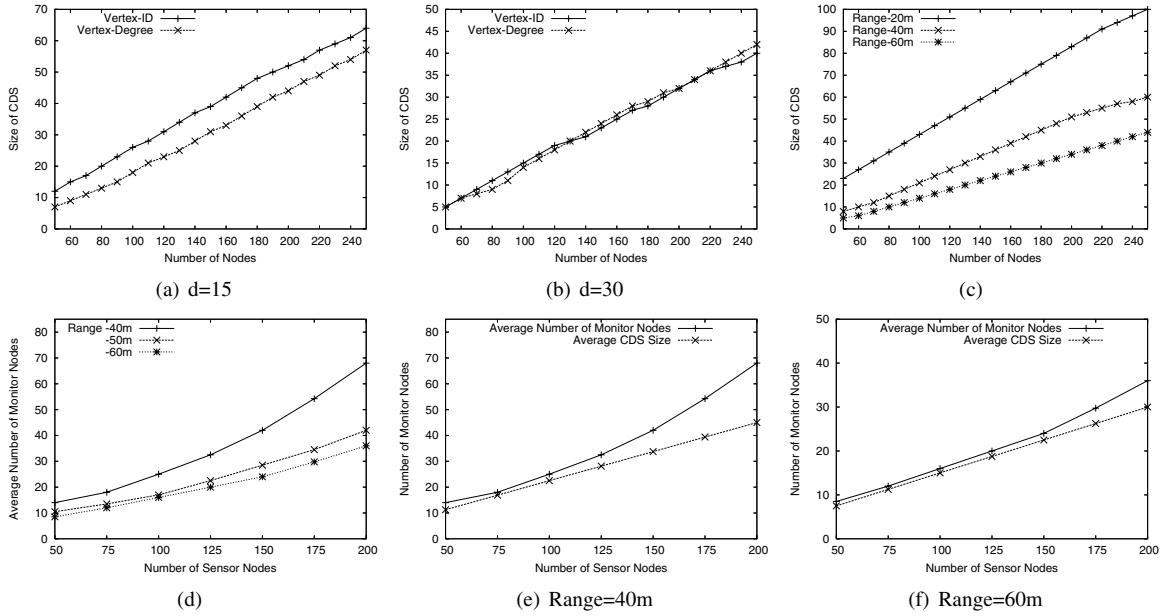
In [6], Brainard et al. introduce the concept of vouching as a tool for on-line authentication. The AVA authentication scheme proposed in [9] is an extension of [6]. AVA has special nodes called Ambassadors, which are selected and dispatched according to several different criteria to represent their home region and perform node authentication. Any node that intends to move into a new region searches for its ambassador in the new region and takes the ambassador's authentication for transferring its reputation. But in AVA the adversary can compromise the ambassador nodes to launch its attacks and the authors have not addressed this problem. Also, AVA is vulnerable to node replication attacks. However, in our model, since tamper-proof monitors are used for authenticating, the adversary has no way of playing foul.

6 Simulation

All simulations have been carried out on a custom built, stand-alone, C++ simulator. In our simulations, a sensor field of area $100 \times 100\ m^2$ has been considered. The number of sensor nodes S, number of monitor nodes M, and transmission range have been considered as tunable parameters. The network has been modeled as an undirected graph.

In Figs. 3 (a) and (b), we have presented results comparing the impact of network diameter on size of the CDS. We have compared the size of CDS size obtained with vertex ID and vertex degree as the priority values for $d = 15$ and $d = 30$ respectively. We can see that with increasing diameter, the size of the CDS shrinks. We have also studied the impact of density on CDS size by fixing the transmission ranges at 20, 40, and 60m and the results are presented in Fig. 3 (c). It is evident that as the transmission range increases, the size of the CDS shrinks. With range fixed at 20, 40, and 60m, the size of the CDS is approximately 42%, 22%, and 15% of the size of the network respectively. In the rest of our simulations, unless otherwise specified, vertex ID is used as the priority value in constructing the CDS.

In Fig. 3 (d), we have presented results comparing the average number of monitor nodes required for varying number of sensor nodes. We can see that as the number of sensor nodes increases, the number of monitor nodes required increases. However, this number decreases with increasing transmission range of monitor nodes. In Figs. 3 (e) and (f), we have presented results comparing the average CDS size for varying numbers of sensor nodes with the average number of monitor nodes required to ensure the existence of a CDS for monitor transmission range of 40m and 60m respectively. We see that the number of monitor nodes required is lower with a monitor range of 60m compared to a range of 40m. Due to space limitations, detailed simulation results will include it in an extended version of this paper.



(a) d=15 (b) d=30 (c)

(d) (e) Range=40m (f) Range=60m

Figure 3. (a) Size of CDS with Fixed d=15 (b) Size of CDS with Fixed d=30 (c) Size of CDS with range varied from 20m to 60m. (d) Average number of monitor nodes for varying numbers of sensor nodes (e) - (f) Average number of monitor nodes required and the average size of CDS.

7 Conclusion

In this paper, we have proposed a novel Connected Dominating Set (CDS)-based reputation monitoring system. Our model is the first attempt to employ a CDS-based monitoring backbone to securely aggregate the reputation of sensors without subjecting them to energy depletion or reputation pollution attacks encountered in existing reputation monitoring systems. Secure and certificateless node mobility and robustness to node replication and ID spoofing attacks are two vital by-products of our model. We have confirmed the validity and performance of our model via simulation studies. In our future work, we plan to conduct a more in depth simulation of our model. We also wish to investigate the possibility of using the CDS property to address other security threats in wireless and ad-hoc networks.

References

[1] A. Josang and R. Ismail. The beta reputation system. *In Proceedings of 15th Bled Electronic Commerce Conference*, 2002.

[2] P. Michiardi and R. Molva. CORE: A COllaborative REputation mechanism to enforce node cooperation in Mobile Ad Hoc Networks. *Communication and Multimedia Security*, 2002.

[3] S. Buchegger and J. Boudec. A Robust Reputation System for P2P and Mobile Ad Hoc Networks. *In Proceedings of Economics of P2P Systems*, 2004.

[4] F. Dai and J.Wu. An Extended Localized Algorithm for Connected Dominating Set Formation in Ad Hoc Wireless Networks. *IEEE Transactions on Parallel and Distributed Systems*, 2004.

[5] S. Ganeriwal and M. Srivastava. Reputation-based Framework for High Integrity Sensor Networks. *In Proceedings of ACM SASN*, 2004.

[6] J. Brainard, A. Juels, R. Rivest, M. Szydlo, and M. Yung. Fourth factor authentication: Somebody you know. *In Proceedings of ACM CCS*, 2006.

[7] A. Srinivasan, J. Wu, and J. Teitelbaum. Distributed Reputation-Based Secure Localization in Sensor Networks. Accepted to appear in a *Special Issue of Journal of Autonomic and Trusted Computing*, 2008.

[8] F. Li and J. Wu. Mobility Reduces Uncertainty in MANETs. *In Proceedings of INFOCOM*, 2007.

[9] F. Li and J. Wu. Authentication Via Ambassadors: A Novel Authentication Mechanism in MANETs. *In Proceedings of MilCom*, 2007.