A game theory based approach for security in wireless sensor networks

Afrand Agah, Sajal K. Das and Kalyan Basu
Center for Research in Wireless Mobility and Networking (CReWMaN)
Department of Computer Science and Engineering
University of Texas at Arlington
Arlington, TX 76019-0015
{agah, basu, das}@cse.uta.edu

Abstract—**Based on cooperative game theory, we propose a new technique for handling security issues in mobile wireless sensor networks. We define a game between sensor nodes and concentrate on three fundamental factors: cooperation, reputation and quality of security. Stronger cooperation between two nodes implies more reliable data communication between them. And the more a node cooperates, the better is its reputation, which decreases when misbehavior is detected. When security of the network is compromised, the percentage of exposed traffic measures the quality of security of sensor nodes. By incorporating these three factors, we cluster the sensor nodes such that within a cluster, the payoff function of all sensor nodes are close to each other, where payoff is the largest possible individual gain for each sensor according to a defined utility metric. We define one strategy set for each node, which guarantees reaching to an equilibrium point for payoff function.**

I. Introduction

It is crucial that the security of sensor networks be monitored and diagnosed to ensure correct behavior. This is challenging in an environment where the network is designed to be flexible. The major challenges in tackling wireless sensor networks security include: power conservation for mobile sensors, cooperation among heterogeneous sensors, flexibility in the security level to match the application needs, scalability, self organizing and self learning capabilities of sensors, trust and security decisions for the application, keeping the mobility and volatility transparent, and yet protecting the network from external and internal intrusions. In a nutshell there are three factors that we have to consider energy, computation and communication.

Due to resource scarcity (battery power, memory, and processing power) of sensors, securing sensor networks is quite different from traditional schemes that generally involve management and safe keeping of a small number of private and public keys [7]. Disclosing a key with each packet requires too much energy [10]. Storing one-way chain of secret keys along a message route requires considerable memory and computation of the nodes on that rout [11]. The key management using a trusted third party requires an engineered solution that makes it unsuitable for sensor network applications [2]. Although the asymmetric key cryptography does not require a trusted server, key revocation becomes a bottleneck [3] as it involves an authority maintaining a list of revoked keys on a server or requesting the public key directly from the owner.

In this paper, we formulate a cooperative game between sensor nodes and propose a novel framework for forming clusters in wireless sensor networks, where each cluster is a cooperative environment and consists of a subset of sensor nodes. We propose a payoff function, which is based on three fundamental issues: cooperation, reputation and quality of security. Then we define the game strategy set such that it guarantees reaching to an equilibrium point for payoff function. One important fact about this game theoretic approach is that we have not considered a fixed value for the total number of possible clusters. By simulation experiments, we compare our approach with the one in which distance (instead of utility function value) is used as the criterion for clustering sensor nodes. We observe that the average number of clusters and total number of message passing are much higher for distance based clustering. The advantage of our game theory framework comes from the fact that we can apply and accumulate the cooperation and reputation in the utility function, while aggregating the quality of security.

The rest of the paper is organized as follows. Section II summarizes related works on security protocols for sensor networks. Section III presents our proposed model, objectives and the algorithm. In section IV we discuss the simulation environment and evaluate our approach. Section V concludes the paper.

II. Related Works

The limited resource of sensor nodes makes it undesirable to use public-key algorithms, such as Diffie-Hellman key agreement [6]. A sensor node may need tens of seconds or even minutes to perform these operations. As sensor nodes are usually deployed in large numbers, it is desired that each sensor be low-cost. Consequently, it is hard to make them tamper-resistant [5].

Different security protocols have been proposed for sensor networks. The SNEP protocol [10] has low communication overhead (only 8 extra bytes per message), providing baseline security primitives like data confidentiality, two-party data authentication, reply protection and message freshness. It achieves semantic security (the same message is encrypted

0-7803-8396-6/04/$20.00 © 2004 IEEE

differently each time), thus preventing eavesdroppers from inferring the content from the encrypted message. The μ TESLA protocol [9] uses a symmetric key mechanism. To generate one-way key chain, the sender chooses the last key randomly and generates the remaining values by successively applying a one-way function. The protocol discloses the key once per time interval (rather than one key per packet), and restricts the number of authenticated senders. To bootstrap, each receiver needs one authentication key of one-way function key chain. The base station can also broadcast disclosed key and performs initial bootstrapping for new receivers to conserve energy. The periodic key disclosure of μ TESLA ensures that compromising a single sensor does not reveal the keys of all the sensors in the network.

A good security design for sensor network must be able to secure node-to-node communications. It also should not involve the base station in each and every articulation. Untrusted nodes should not be able to stay in the network for a long time. Nodes should not have any a priori knowledge about which other nodes are close enough in their transmission range. And it is very likely that the topology of the network changes very frequently. To the best of our knowledge, no reported work uses game theory for clustering sensor nodes taking into account all the above aspects. In the following section we describe how to design such a framework.

III. Our proposed model

Due to the resource constraints, a sensor node in our algorithm does not need to have information about other sensors in the network. We consider a fully dynamic network and all communication between clusters is through clusterheads. Moreover, in our network every node can be malicious, and isolating malicious nodes is one of our goals.

A. Game and Payoff function

Our cooperative game is defined as: $\Gamma =< I, S, U >$, where I is the set of sensor nodes (players), $S = \{S_i\}$, where S_i is the set of strategies for node $i \in I$ and $U = \{U_i\}$, and U_i is the payoff function for node i. Our goal is finding the largest payoff that sensor nodes within a cluster can collectively achieve. We define an optimal payoff function that fulfills our objectives for securing a sensor network. This function consists of three factors: (i) cooperation, (ii) reputation and (iii) quality of security.

The payoff between two sensor nodes should be dependent on their distance and each node's transmitter signal strength. The more the transmitter signal strength, the more likely the node cooperates with its neighbors. A far-field region is the region where the angular field distribution is essentially independent of distance from the source [8]. If the source has a maximum overall diameter D, which is large compared to the wavelength λ, the far-field region is commonly taken to exist for distances greater than $2D^2/\lambda$ from the source. The far-field region is also referred to as the Fraunhofer region. The distance, d, of a sensor (transmitter) from another sensor (receiver) is derived from the measurement of received signal strength (in dBm) as: $Q_{ij}(t) = \frac{T(t)}{d_{ij}^2}$, where $T(t)$ is the transmitter signal strength (in dBm). Formally, the cooperation, between two sensor nodes at time t is defined as:

$$\Omega_{ij}(t) = \begin{cases} Q_{ij}(t) + f - \beta' & for \quad d_{ij} < D \\ f & for \quad d_{ij} > D \end{cases}$$

where the factor f stands for minimum signal strength (in dBm). As the channel bandwidth is finite, we consider that if a node plays cooperatively (forwards more packets), it must bear some additional cost β'. Sensor nodes belonging to one cluster can exchange information flexibly to yield certain comparable payoffs.

In order to show how much each individual sensor node is useful for the whole network, the payoff between two nodes should also represent how many packets each node receives and forwards at each time slot. Let $P_{ij}^f(t)$, $P_{ij}^g(t)$ and $P_{ij}^r(t)$ be respectively the number of packets forwarded, generated, and received between two sensor nodes, i and j at time t. Each of these random variables represents the total number of occurrences of some phenomenon during a fixed period of time. We assume that the number of occurrences in any two disjoint intervals of time is independent of each other. Also, the probability of an occurrence during very short interval of time is approximately proportional to the length of that interval, so its distribution is Poisson.

We define the reputation, $\Phi_{ij}(t)$, as the ratio of the number of packets forwarded to the total number of received and generated packets between two nodes at time t. Thus $\Phi_{ij}(t)$ is measure of throughput experienced between every two nodes. Formally,

$$\Phi_{ij}(t) = \gamma \frac{\sum_{ij} P_{ij}^f(t)}{\sum_{ij} P_{ij}^r(t) + \sum_{ij} P_{ij}^g(t)}$$

where $0 < \gamma < 1$ accounts for misbehavior potential due to such factors as past record of mobile code handling and potential tampering of sensor hardware. The reputation value decreases when misbehavior (security violation) is detected. Note that $\Phi_{ij}(t) < 1$.

The payoff between two sensors should also represent trustworthiness of the traffic. We define the *quality of security*, $\Theta_{ij}(t)$, for each cluster as the percentage of exposed traffic, if security is compromised. Let $\sum_{ij} M_J^g(t)$ be the total number of messages generated between nodes i and j, that belong to a specified cluster J, during time interval t. Let $\sum_{ij} M_J^d(t) \geq 0$ denote the total number of messages dropped between nodes i and j during time t in cluster J. The difference between total number of messages generated between two nodes and total number of messages dropped between them is the number of messages that have been exposed in cluster J but not transferred to the destination, due to the untrustworthiness of destination, and indicates bad reputation or low cooperation between source and destination. The quality of security in cluster J during time t is given by:

$$\Theta_{ij}(t) = \frac{\sum_{ij} M_J^g(t) - \sum_{ij} M_J^d}{\sum_{ij} M_J^g(t)}.$$

For simplicity, we define the payoff utility function, $U_{ij}(t)$, as linear combination: $U_{ij}(t) = \alpha\Omega_{ij}(t) + \beta\Phi_{ij}(t) + \delta\Theta_{ij}(t)$,

TABLE I

PARAMETERS & NOTATIONS

Parameter	Notation
Reputation	$\Phi_{ij}(t)$
Cooperation	$\Omega_{ij}(t)$
Quality of Security	$\Theta_{ij}(t)$
Strategy of node i	s_i
Number of strategies	m
Traffic between nodes	$\tau_{ij}(t)$
Utility between nodes	$U_{ij}(t)$
Distance between nodes	$d_{ij}(t)$
Probability of doing a strategy	p_i
Total number of messages dropped between nodes	$\sum_{ij} M_j^d(t)$
Total number of messages generated between nodes	$\sum_{ij} M_j^g(t)$

α, β, δ, are weight parameters and $\alpha+\beta+\delta = 1$. Depending upon the sensor applications, their value can be varied. Table 1 depicts parameters and their notations which have been used throughout this paper.

B. The Game Strategy set

Now that the set of players and the payoff functions are defined, let us define a strategy set. In each time slot, each sensor node uses its strategy based on the information it obtained in preceding time slots according to three following criteria: (i) reputation (Φ_{ij}): sensor nodes have not made enough reputation to trust each other and cooperate with each other, (ii) distance (d_{ij}): the closer the two nodes, the more they trust each other and (iii) traffic (τ_{ij}): sensor nodes do not have a good history of joint operation and will not trust each other. If there is enough reputation level, closeness of sensor nodes and good history of joint operation, then the strategy for sensor nodes is to cooperate with each other by receiving or forwarding incoming packets; and otherwise the strategy is to defect.

For establishing boundaries in order to choose a particular strategy, we define $\nabla_{ij}(t) = (\Phi_{ij}(t-1), d_{ij}(t-1), \tau_{ij}(t-1))$. Let $T = [0, t]$, then we define $\chi^{ij} = \{\nabla_{ij}(t)$ for $t \in T\}$, $\chi^{ij}_{min} = \{\nabla_{ij}(t)$, such that $||\nabla_{ij}(t)||$ is minimum for $t \in T\}$, and $\chi^{ij}_{max} = \{\nabla_{ij}(t)$, such that $||\nabla_{ij}(t)||$ is maximum for $t \in T\}$. Now we define the strategy as the probability of cooperation:

$$S^m_{ij}(t) = \begin{cases} p_1 & if \quad m = 1, \quad \nabla_{ij}(t) \in \chi^{ij}_{min} \\ p_2 & if \quad m = 2, \quad \nabla_{ij}(t) \in \chi^{ij} - (\chi^{ij}_{min} \cup \chi^{ij}_{max}) \\ p_3 & if \quad m = 3, \quad \nabla_{ij}(t) \in \chi^{ij}_{max} \end{cases}$$

where $p_1 < p_2 < p_3$, $p_1 + p_2 + p_3 = 1$. Between every two nodes three possible strategies exist, for example $S^1_{ij}(t)$ represents the first strategy of node i against node j.

At each time unit the new values of the payoff utility function for each node are calculated. For sensor nodes i, j and k, where $i \neq j \neq k$, if $|U_{ij}(t) - U_{ik}(t)| < \epsilon$, these nodes will belong to one cluster. Certain nodes, known as ***clusterheads*** will be chosen in each cluster, which are responsible for communicating with other clusterheads. As sensor nodes move, they can leave their original cluster and join another cluster. If they are out of radio range of any existing clusterheads, then new clusters can be formed. A clusterhead can also change its position. Thus sensor nodes can change the cluster partnership, and clusterheads are formed and deleted dynamically.

Next, we show how to reach equilibrium and investigate conditions to achieve them. Given a game Γ, utility function U_t and strategy s_i, an n-tuple strategy $(s_1, s_2, ..., s_n)$ is said to be in equilibrium, if and only if, for any $i = 1, ..., n$, and any strategy s^*_i, $U_t(s_1, ..., s_{i-1}, s^*_i, s_{i+1}..., s_n) \leq U_t(s_1, s_2, ..., s_n)$. So the n-tuple of strategies is said to be in equilibrium if no player has any positive reason for changing the strategy. Sensor nodes aim for maximizing the payoff function chosen accumulated over time, which is completely dependent on the strategies. In order to find the equilibrium point of the game while considering two nodes i and j, we ponder two cases. (i) when two nodes are in two different clusters and (ii) when two nodes are in the same cluster. What we need to find is the strategy set that maximizes the value of $U_{ij}(t)$.

Theorem 1: Game has an equilibrium at strategy pair $\{S^3_{ij}, S^3_{ji}\}$.

Proof: Case I: As both nodes are in different clusters, the value of $\Theta_{ij}(t)$ is negligible since $\sum_{ij} M^g_j(t) = \sum_{ij} M^d_j = 0$. Also $Q_{ij}(t) = \frac{\tau_{ij}(t)}{d^2_{ij}}$ is a small value due to the fact that d^2_{ij} is large. The final value of payoff will be dependent to the value of $\Phi_{ij}(t)$. On the other hand, the total number of generated packets is constant, and the total number of forwarded and received packets are dependent on the strategy, which indicates the probability of these events. Let us consider function $f : N \to N$ such that $f(k) = S^k_{ij}(t)\Sigma_{ij}P^f_{ij}(t)$, which denotes the total number of forwarded packets with probability $S^k_{ij}(t)$. It has the following property: if $k \geq k'$, then $f(k) \geq f(k')$. Since $f(k)$ is monotonically increasing, it attains maximum value at $k = 3$.

Case II: As both nodes are in the same cluster, the value of $Q_{ij}(t)$ is constant. The total number of generated packets and $\sum_{ij} M^g_j$ are also constant. As in the above case, suppose $f(k) = S^k_{ij}(t)\Sigma_{ij}P^f_{ij}(t)$. Similarly we can define the total number of dropped packets in cluster J with probability $p_k = S^k_{ij}(t)$, with a function g, such that $g(1 - p_k) = p_k \sum_{ij} M^d_j$. Therefore, the final payoff is dependent on $f(k) + g(1 - p_k)$. Note that f is monotonically increasing and $f(k) \geq g(k)$ (i.e., total number of forwarded packets is greater than total number of dropped packets), g can be expressed as $g(p_k) = \alpha' f(k)$ where $0 < \alpha' < 1$. So we conclude that p_3 maximizes the value of $f(k) + g(1 - p_k)$. ■

IV. SIMULATION & EVALUATION

We simulate a system of N sensor nodes, each with location (x_i, y_i) for $1 \leq i \leq N$. All nodes can move in all possible directions, in each time unit and are uniformly distributed in a 500 × 500 square meter area. The simulation consists of the following steps: (i) randomly distribute sensor nodes in a grid space, (ii) compute cooperation, reputation and the quality of security for each node, (iii) compute the payoff function value for all nodes, (iv) define clusters based on the payoff function, (v) define a clusterhead for each individual cluster, (vi) randomly move all nodes including

clusterheads, (vii) reorganize clusters. One important fact about this approach is that we have not considered a fixed value for the total number of clusters. We compare our results with the case in which distance is used instead of utility function as the criterion for clustering sensor nodes.

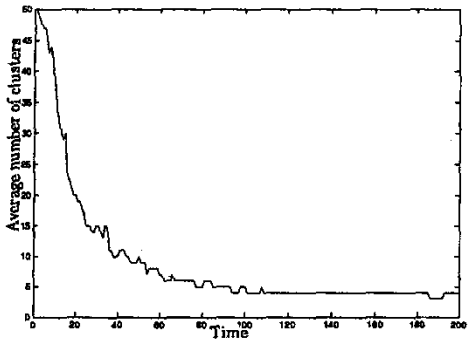

Fig. 1. Average number of clusters for $\epsilon = 0.2$

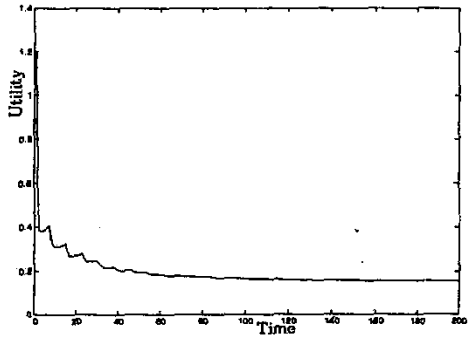

Fig. 2. Payoff function

In our simulation experiments, N was varied between 5 and 200. Total time of simulation was up to 200 seconds. The parameters used in simulation were $\alpha = 0.3$, $\beta = 0.3$, $\delta = 0.4$, $\epsilon = 0.15$, $p_1 = 0.1$, $p_2 = 0.3$, $p_3 = 0.6$, $\chi_{min} = 0.2$ and $\chi_{max} = 0.7$. At every time unit, all sensor nodes move with a velocity that is uniformly distributed with average of 0.1 meter/second. New values for x and y coordinates of each node are calculated as: $x_{t+1}(i) = x_t(i) + \Delta(t) \times v_x \times \lambda$ and $y_{t+1}(i) = y_t(i) + \Delta(t) \times v_y \times \lambda$. Here λ is a random number $0 < \lambda < 1$ and v_x and v_y are velocity of each individual sensor node.

In order to show the effectiveness of our clustering approach using utility function, we compare it with the existing distance based clustering approaches. In the following figures, the legend "utility" (payoff) means the results obtained from the proposed algorithm and "distance" means the results obtained based on distance between nodes, under the same simulation environment with the same set of parameters.

Fig. 1 shows the average number of clusters, in the beginning it has too much oscillations but around $t = 60$, which is where the payoff becomes stable, it does not have too much oscillations. This shows, sensor nodes reach a stability within the clusters. It is not completely stable, which is due to the random movement of sensor nodes. The higher the value of ϵ, the larger neighborhood and the lower amount of clusters. Results indicate that it is very important to choose the right parameters for cluster formation. For very large values of ϵ, all sensor nodes belong to only a few clusters, which is undesirable as the clusterhead must try to serve more sensor nodes than it is capable of, so system efficiency will suffer too.

Fig. 2 depicts the average value of payoff function. Although all sensor nodes including the clusterhead can move, in order to save energy and computation time, it is better to keep the previous clusterheads as new clusterheads as long as possible. After initialization of the network and the first movement of sensor nodes, we determine clusters and choose one clusterhead for each cluster. In order to determine clusterheads we adapted the weight clustering approach (WCA) described in [4]. Due to movements of sensor nodes and clusterheads, we need a mechanism to reorganize the clusters. Four possible scenarios may occur: (i) if the sensor is out of the radio range of existing clusterhead but within the range of another clusterhead, transfer the sensor node to the later cluster, (ii) if the clusterhead is within the radio range but it covers the range of existing clusterhead, the clusterhead must join the existing cluster and is no longer a clusterhead by itself, (iii) if the sensor is out of range of existing clusterhead and out of range of any other existing clusterhead, then define new cluster, and (iv) if the clusterhead is out of range of an access point, then drop the cluster and define a new cluster.

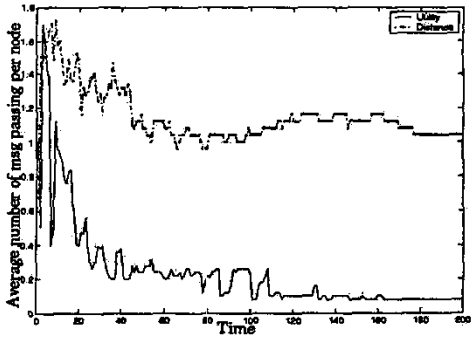

Fig. 3. Average Number of message passing per node vs time

Fig. 3 depicts the average number of message passing per node versus time, when the utility and distance are used as the clustering criteria. The graph increases at the beginning and then saturates as the total number of clusters reaches a stable point. However, notice that the total number of messages is much less for the utility based approach. This is because the distance based approach results in more clusters and hence the average number of message passing is also higher.

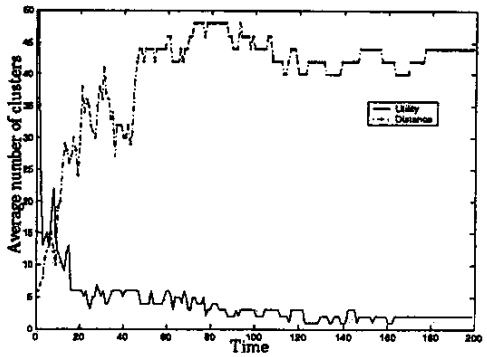

Fig. 4. Average Number of clusters for $\epsilon = 0.25$

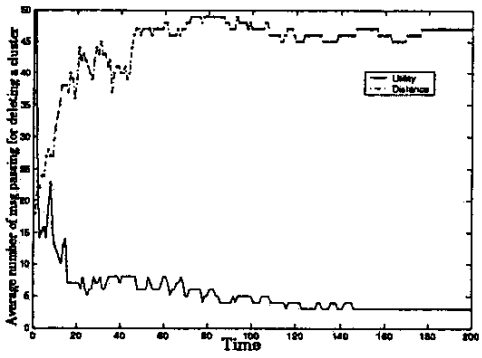

Fig. 5. Average number of message passing for deleting clusters

Fig. 4 depicts the average number of clusters. It shows, if the distance is used as the clustering criterion then more number of clusters are needed. On the other hand, the utility based approach has the distance factor embedded in it, and it also incorporates the reputation of sensor nodes. So if a sensor node has been a good member of the network then other nodes prefer to be in its cluster. It also incorporates the quality of security, so if a sensor node has been trustworthy in the past, other nodes prefer to be in its cluster. In order to justify the performance of this approach, we count the number of message passing that this protocol needs. Figures 5 and 6 respectively depict the average number of message passing for deleting a cluster and deleting a node from a cluster. Also, these values became stable. The distance based clustering has more number of message passing between sensor nodes. Nodes do not join previously shaped clusters and instead they form new ones, which implies more number of clusters and message passing between them. As the results depict, it never reach stability. For utility based clustering the number of messages needed for shaping a new cluster will become stable and no more new clusters will be formed.

V. Conclusion

We investigated a game theoretic framework for securing sensor networks. A new method for clustering is proposed

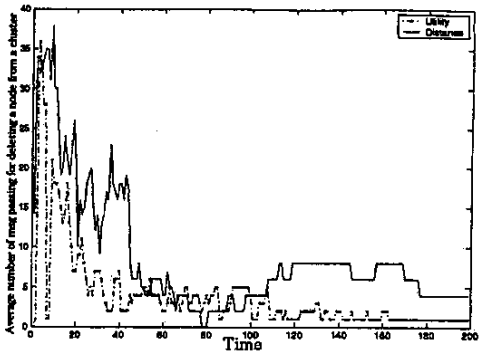

Fig. 6. Average number of message passing for deleting a node from a cluster

which is based on the payoff function. Our current focus was on two main issues: (i) stability of the payoff function with respect to the movement patterns of the sensors, and (ii) optimal parameter settings of cluster formation to reduce the message passing overheads. By choosing appropriate cooperation formula, we showed how to conserve more power for sensors nodes and thus have cooperation among heterogeneous sensors. We also built trust and security decisions based on the payoff function between sensor nodes, while keeping the mobility and volatility transparent. Formulating a non-cooperative game between the network and the intruder is our next goal. Another goal is how to enhance the payoff function from linear to an exponential function.

VI. Acknowledgment

This work is supported by NSF ITR grants IIS-0326505 and IIS-0121297.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey", *Computer Networks*, Vol.38, pp: 393-422, 2002.

[2] L. Buttyan, J. P. Hubaux and S. Capkun, "A Formal Analysis of Syversons Rational Exchange Protocol" , *CSFW*, June, 2002.

[3] S. Capkun, L. Buttyan, and J. P. Hubaux, "Self-Organized Public Key Management for Mobile Ad hoc Networks", *MobiHoc*, 2002.

[4] M. Chatterjee, S. K. Das and D. Turgut, "WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks", *Cluster Computing* , Vol.5, Kluwer Academic Publishers, pp:193-204, 2002.

[5] H. Chan, A. Perrig and D. Song, "Random Key predistribution Schemes for Sensor Networks", *Proceedings of the 2003 IEEE Symposium on Security and Privacy(SP'03)*, 2003.

[6] W. Diffie and M. E. Hellman, "New directions in cryptography", *IEEE Trans. Inform.Theory*, pp:644-654, November 1976.

[7] A. Menezes, P. Oorschot and S. Vanstone, "Handbook of Applied Cryptography", CRC Press, 1997.

[8] A. Papoulis, "Signal analysis", *The McGraw Hill Inc*, New York, 1977.

[9] A. Perrig, R. Canetti, J. Tygar and D. Song, "Efficient Authentication and Signing for Multicast", NDSS, 2001.

[10] A. Perrig, R. Szewczyk, V. Wen, D. Culler and J. D. Tygar, "SPINS: Security Protocols for Sensor Networks", *ACM MobiCom*, pp: 189-199, July 2001.

[11] A. Perrig and J. D. Tygar, "Secure Broadcast Communication in Wired and Wireless Networks", *Kluwer Academic Publisher*, 2003.

[12] N. N. Vorobev, Game Theory Lectures for Economists and Systems Scientists, *Springer-Verlag*, 1977.