A Cluster-based Countermeasure against Media Access Control Layer Attacks in IEEE 802.11 Ad Hoc Networks

Fei Shi[1] and JooSeok Song[2]
1,2 Department of Computer Science, Yonsei University
Shinchon-dong, Seodaemoon-gu, Seoul, 120-749, South Korea
[e-mail: { shifei,jssong}@emerald.yonsei.ac.kr]
*Corresponding author: JooSeok Song

Received February 20, 2011; revised May 22, 2012; accepted June 13, 2012; published June 25, 2012

Abstract

The characteristics of ad hoc networks, such as the absence of infrastructure, a dynamic topology, a shared wireless medium and a resource-constrained environment pose various security challenges. Most previous studies focused on the detection of misbehavior after it had occurred. However, in this paper we propose a new way of thinking to evade the occurrence of misbehavior. In our scheme, we firstly present a clustering algorithm that employs a powerful analytic hierarchy process methodology to elect a clusterhead for each cluster. The clusterhead in each cluster is then allowed to assign the backoff values to its members, i.e., originators, rather than permitting the originators to choose the backoff values by themselves. Through this media access control layer misbehavior detection mechanism, the misuse of the backoff in the media access control layer in the 802.11 distributed coordination function can be detected.

Keywords: MAC, ad hoc network, AHP, clusterhead, backoff

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1B3004161).

http://dx.doi.org/10.3837/tiis.2012.06.005

1. Introduction

Compared with wired networks and certain types of wireless networks, ad hoc networks are more vulnerable to security attacks because of their unique characteristics. Examples of attacks include the rushing attack, blackhole attack, Media Access Control (MAC) layer attack and so on [1][2]. This paper focuses on the detection of misbehavior in the MAC layer resulting from the misuses of the backoff mechanism.

The MAC layer protocol [3] provided in the IEEE 802.11 family of standards was designed to establish cooperation among nodes in the networks. This protocol assumes that all nodes behave properly and actively. However, attackers could violate MAC layer protocol simply by misusing the backoff value. Thus, the development of mechanisms to handle the MAC layer misbehavior is essential. In this paper, we propose a scheme for the detection of any misbehavior in the MAC layer in the 802.11 Distributed Coordination Function (DCF). The key idea is to avoid any MAC layer misbehavior attack by setting the backoff value to the originator and then monitoring whether the originator obeys the backoff.

This paper is organized as follows: In Section 2, we describe the model of the MAC layer misbehavior attack and then summarize and discuss the related research. In Section 3, we present the details of our scheme. First, we present a clustering algorithm to elect clusterheads (CHs) using the Analytic Hierarchy Process (AHP) methodology. We then present a mechanism for the detection of any misuse in the backoff stage of 802.11 DCF. In Section 4, we present the discussion and future work. Finally, we conclude the paper in Section 5.

2. Related Work

The DCF of 802.11 specifies the use of Carrier Sense Multiple Acess with Collision Avoidance (CSMA/CA) to decrease collisions in wireless networks. A node that intends to transmit packets picks a random backoff value in between $[0, CW]$, where CW is the contention window size, and then performs transmission after waiting for the backoff value delay. Nodes exchange Request to Send (RTS) and Clear to Send (CTS) packets to reserve the channel before transmission. Other nodes that overhear either the RTS or the CTS are required to defer transmissions on the channel during the conservation period. If a transmission is unsuccessful, the CW value is doubled. If the transmission is successful, the node resets its CW to a minimum value CW_{min}.

However, a misbehaving node may attack the MAC layer in several ways. One method is the selection of backoff values from a different spectrum that has average backoff values that are smaller than the backoff values specified by the DCF in 802.11.

For example, selecting backoff values from the range $\left[0, \frac{CW}{2}\right]$ instead of $[0, CW]$ results in a higher probability of possessing the medium, or selecting backoff values from the range $[CW, 2CW]$ instead of $[0, CW]$ induces selfish nodes to reduce their consumption and resource. Another method involves the use of a different retransmission strategy that does not double the CW value after collision, as specified by the DCF in 802.11; for example, tripling the CW value or multiplying the CW value by 1.5. The tripling retransmission strategy can make the corresponding node selfish. Hence, the selfish node will not participate in the network as expected. Conversely, the latter can geive a greater opportunity to consume a larger bandwidth.

Misbehavior in the MAC layer has been primarily addressed based on the game theory. To guarantee that the network reaches equilibrium, Cagalj et al. specify the mechanism that each node should follow in terms of controlling channel access probability by adjusting the contention window using a dynamic game model [4]. They present the conditions in which the Nash equilibrium of the network with several misbehaving nodes is still Pareto-optimal for each node. The problem with this scheme is that it assumes that all nodes are within the wireless range, i.e., all nodes can communicate with one another directly. However, this assumption is not always valid in practical ad hoc networks.

Another way of considering the same problem in the MAC layer is provided by Kyasanur et al. [5]. They propose a modification to IEEE 802.11 to detect misbehaving nodes. In their scheme, the receiver assigns the backoff value to the originator, such that the receiver can detect any misbehavior of the originator. The problem with applying this protocol to ad hoc networks is that the receiver might not be trustworthy, because each node in ad hoc networks works both as a terminal and a router and has equalized security status. The trustworthiness of the receiver cannot always be guaranteed.

In our previous paper [10], we proposed a MAC layer misbehaviors avoidance and detection mechanism. We introduced the local most trustworthy (LMT) node. The LMT node was defined as the node that owns the largest trust value in the neighborhood of the originator node that was required to set backoff values to the originator. Accordingly, a trust management mechanism was proposed to assign the trust value for each node. The trust value was defined as a function of two parameters: credit value and stability value. The scheme works well in small-scale ad hoc networks in which the number of nodes is small, because this scheme can be described as reactive, i.e., whene an originator wants to send some packets, the corresponding LMT node is needed reactively. Thus, the subsequent overhead is relatively low. However, when the number of originators increases, the number of control messages generated by the election of corresponding LMT nodes also increases which increases the overhead significantly. Thus, the scalability cannot be guaranteed for a large-scale ad hoc network in [10].

3. Proposed Scheme

In this paper, we propose a novel scheme for the detection of MAC layer misbehavior by preventing the attacker's misuse of the backoff value. The basic scheme is described as follows: Instead of allowing the originator to select the backoff values by itself to initialize the backoff counter, the backoff selection is performed by the CH [11], which is the coordinator of the cluster in which the originator resides. The originator may then use this backoff value as its initial backoff counter for the following transmission. Meanwhile, the CH continues monitoring the originator's compliance with the backoff counter provided by the CH. The implementation of monitoring can be achieved using the watchdog mechanism [6]. By comparing the expected backoff value selected by the CH and the actual backoff value used by the originator, the CH can determine whether the originator is a MAC layer misbehavior attacker.

This process gives rise to two issues. The first issue is the selection of the CH, and the second is the method by which to determine whether the originator is an attacker. We propose a clustering algorithm to address the first issue. This algorithm uses the AHP methodology [9] to calculate the weight of each node in an ad hoc network.

3.1 Preparatory Work

In this subsection, we first introduce the evaluation of three parameters that together determine the weight of each node. The three parameters are relative stability (S_r), credit value (C_v), and the reciprocal of the forward rate (R_f). The first parameter S_r is evaluated based on the rate of neighbors' change. S_r is a relative value, as implied by its name. S_r indicates whether the node moves relatively fast or slow, or whether the node keeps stable compared with its neighbors. The second parameter C_v is evaluated based on the transmission behavior of the nodes. C_v indicates whether the nodes exhibit misbehavior by dropping packets (to simplify and clarify the problem, we only consider the dropping of packets to be the misbehavior, aside from any misbehavior in the MAC layer). The third parameter R_f is evaluated based on the packet forwarding rate. R_f indicates whether the nodes violate the aforementioned backoff mechanism specified by the DCF in 802.11. R_f also indicates the remaining battery power of each node because frequent packet forwarding could enhance the rate of battery power consumption. In summary, a node that remains relatively stable, or that moves slowly and exhibits normal behavior with a lower forwarding rate has a greater probability of being elected as the CH. Conversely, the node that moves fast has a smaller trust value or that performs malicious behaviors (i.e., dropping packets), or with a larger forwarding rate has the least probability of being selected as the CH in the corresponding vicinity.

In the following subsections, we depict each parameter's function and significance, and we present the evaluation procedure.

3.1.1 Evaluation of Relative Stability (S_r)

An explanation of the need for the stability value is needed. Consider the following scenario: One node is moving extremely fast that, such other nodes cannot connect to and communicate with it. In this case, this node is useless and cannot be assigned an important position. In our scheme, we intend to choose a relatively stable CH that could stay in the neighborhood for a longer time, over a node that has a high mobility rate. Considering this factor, the stability value is introduced to evaluate the stabilization of each node.

To provide a reasonably description of node stability, our proposed approach uses the graph theory [7] and a similarity computation method [8]. The network formed by nodes and links can be represented by a directed graph, $G(t) = (V, E(t))$, called the neighbor relation graph, wherein $V = \{1,2,\dots,N\}$ denotes the set of participating nodes, and $E(t) = \{e_1, e_2, \dots, e_m\}$ denotes the set of wireless links. If node i can receive information sent from j, a directed edge $e(i,j)$ exists between node i and node j, i.e., node j is the neighbor of node i. $E_i(t_j)$ and $E_i(t_{j+1})$ are vectors that denote the wireless links situation of node i at two consecutive time points, t_j and t_{j+1}, as shown in Figure 1. According to the similarity theory, the stability value can be represented by the mean similarity value between $E_i(t_j)$ and $E_i(t_{j+1})$ as shown in Equation 1.

$$S_i = \frac{1}{n-1}\sum_{j=1}^{n-1}\cos\theta_j = \frac{1}{n-1}\sum_{j=1}^{n-1}\left[\frac{E_i(t_j)\cdot E_i(t_{j+1})}{|E_i(t_j)||E_i(t_{j+1})|}\right] \quad (1)$$

where θ_j is the included angle between the vectors $E_i(t_j)$ and $E_i(t_{j+1})$, as shown in Figure 2, whereas n is number of time points at which $E(t)$ is observed.

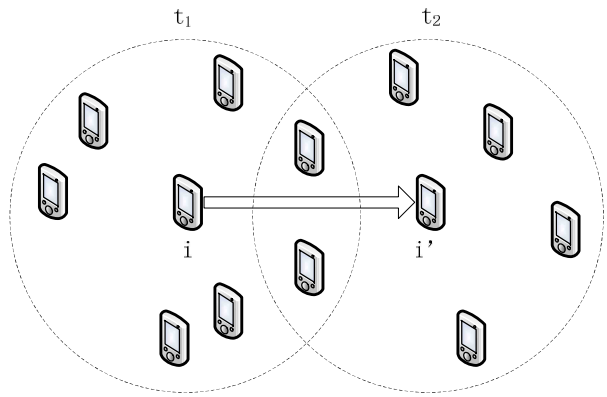

Fig. 1. Scenario showing the change of node i's neighborhood because of mobility

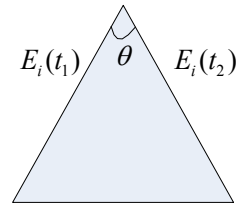

Fig. 2. Intuitive graph representing the similarity between two vectors $E_i(t_1)$ and $E_i(t_2)$

S_i denotes the similarity between node i's vicinity situations status at different time points, e.g., t_j and t_{j+1}. If S_i is larger, the angle θ_j between $E_i(t_j)$ and $E_i(t_{j+1})$, i.e., θ_j, will be smaller. This condition implies a greater degree of similarity between $E_i(t_j)$ and $E_i(t_{j+1})$, i.e., the neighbors of node i do not change dynamically at time points t_j and t_{j+1}, thus indicating that node i is relatively stable. On the contrary, if S_i is smaller, θ will be larger. This condition expresses less similarity between $E_i(t_j)$ and $E_i(t_{j+1})$, i.e., the neighbors of node i change dynamically at time points t_j and t_{j+1}, thus indicating that node i moves fast. Hence, in our scheme, S_i is used to represent the stability of node i.

The main characteristic of an ad hoc network is its dynamic topology; therefore, an adaptation of the algorithm must be available to support this topology. The CH should undergo the least possible change as it moves; hence, a slowly moving node is chosen as the CH; otherwise, the cluster may be broken. Thus, nodes with lower mobility are favored for the role of CHs because the changes in the CHs will be fewer.

3.1.2 Evaluation of Credit Value (C_v)

Each node has two types of credit value. First, when a node can directly observe the behavior of another node, a direct credit can be established. $C_d(i,j)$ denotes direct interactions between nodes i and j. Node i can monitor the behavior of node j and then evaluate the credit value of node j. As shown in Figure 3, node A can directly observe the credit value of node B and E, and then obtain their direct credit values. Second, when a node receives recommendations about one node from other nodes, a recommended credit can be established. Recommended credits are of two types. In one type of recommended value, no direct interaction exists between nodes i and j; however, an indirect interactions may exist between them if the following conditions hold true: (1) an intermediate node k exists between nodes i and j; and (2) interactions exist between nodes i and k and between nodes j and k. Thus, node i can obtain the credit value of j through k. Nodes A and C are not immediate neighbors, but node A can obtain the credit value of node C through the recommendation of node B. In another type of recommended credit value, a direct interaction between exists nodes i and j, and node i can obtain both the direct value and the recommended credit value of node j. Node A can obtain the direct credit value of E, whereas node B can rely the recommended credit value of E to A. In this case, node A

will obtain the recommended value of E through B. The recommended credit value is represented by $C_r(i,j)$.

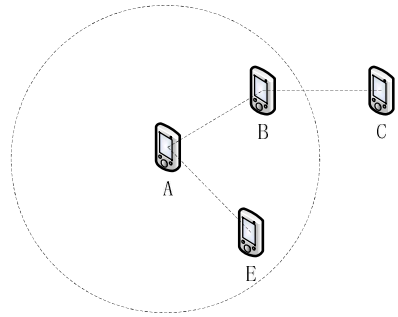

Fig. 3. Scenario of neighbors' connectivity in an ad hoc network

The total credit value $C_v(i,j)$ can be obtained by integrating $C_d(i,j)$ and $C_r(i,j)$ using Equation 2:

$$C_v(i,j) = \omega_1 C_d(i,j) + \omega_2 C_r(i,j) \quad (2)$$

where ω_1 and ω_2 denote the weight factors for $C_d(i,j)$ and $C_r(i,j)$ respectively. We adopt $\omega_1 > \omega_2$, and $\omega_1 + \omega_2 = 1$. We adopt a factor ω_1 that is larger than ω_2 because we consider the direct trustworthiness of one node to be more reliable than the recommended trustworthiness from other nodes. Malicious nodes may provide a dishonest recommendation; hence, the recommended credit value should be treated separately from the regular direct credit value. Thus, we set the factor ω_2 as a relatively smaller value to make it less important. Even if a dishonest recommendation were made, the damage caused by the recommended credit value will be minimal.

The frequently exchang of recommended credit value between nodes will definitely result in increased traffic, and the opportunity for transmission collision will be increased. C_r could be recommended only if it is larger than the threshold to decrease the traffic in the network and avoid congestion. If not, the recommendation is useless and ignored so that the traffic caused by the recommendation will be decreased, and congestion will be avoided.

To simplify and clarify the issue, we assume that the misbehaving nodes only drop packets and do not modify the content of the packets. We consider that the nodes in the network not only share the medium fairly, but also perform their obligations actively. Thus, we consider the dropping of packets as a misbehavior. For example, nodes that drop packets to cut off the network are considered malicious nodes, and other nodes that drop packets to save their energy are likewise considered malicious. The corresponding credit value of the node that drops packets is smaller. The direct credit value (C_d) is established based on whether the previous interactions between nodes i and j are

successful. In other words, $C_d(i,j)$ is node i's evaluation of node j by directly monitoring the packet communication of node j. $C_d(i,j)$ can be calculated by node i using Equation 3. The corresponding parameters are interpreted in Table 1.

$$C_d(i,j) = \frac{N_j^{act}}{N_j} = \frac{N_j^{out} - N_j^{src}}{N_j^{in} - N_j^{dest}} \quad (3)$$

Table 1. Parameters for evaluating direct credit value (C_d)

Number of Packets	Explanation
N_j^{act}	Number of packets actually forwarded by node j
N_j	Number of packets to be forwarded by node j
N_j^{out}	Number of packets that come out of node j
N_j^{src}	Number of packets with node j as the source
N_j^{in}	Number of packets that go into node j
N_j^{dest}	Number of packets with node j as the destination

Equation 3 measures node j's capability to forward packets. Based on the packet transmission direction, two types of packet are related to each node. One type is the packet that "goes into" the node (the number of this type of packet is represented by N_j^{in}); another type is the packet that "comes out" of the node (the number of this type of packet is represented by N_j^{out}). Moreover, the former type of packet ("go into" packet) is divided into two subtypes. One type of "go into" packet is that with node j as the destination (the number of this type of packet is represented by N_j^{dest}). This type of packet should not be forwarded because the destination is node j. Another type of "go into" packet is that which should be forwarded by node j (the number of this type of packet is represented by N_j). Hence, by subtracting the number of packets with node j as the destination (N_j^{dest}) from the number of packets that "go into" the node (N_j^{in}), the number of packets to be forwarded by node j is obtained (N_j). Furthermore, the "come out" type of packet is also divided into two subtypes. One type of "come out" packet is that with node j as the source (the number of this type of packet is represented by N_j^{src}). This type of packet is not forwarded but rather generated by node j. Another type of "come out" packet is that which is actually forwarded by node j (the number of this type of packet is represented by N_j^{act}). By subtracting the number of packet with node j as the source (N_j^{src}) from the number of packets that "come out" of the node (N_j^{out}), the number of packets to be forwarded by node j is obtained (N_j^{act}).

3.1.3 Evaluation of Reciprocal of Forward Rate (R_f)

As mentioned in Section 2, a misbehaving node may attack the MAC layer by selecting backoff values from a different spectrum that has smaller average backoff values than those specified by the DCF in 802.11. This misbehavior increase the likehood of unfairly consuming a larger bandwidth. In this paper, we use the parameter R_f to indicates whether the nodes violate the aforementioned backoff mechanism specified by the DCF in 802.11. Moreover, more frequent packet forwarding results in higher battery power consumption. Thus, we also use this parameter to indicate the remaining battery power in each node. This parameter is evaluated based on the packet forwarding rate, as shown in Equation 4.

$$R_f = \frac{1}{F_r} = \frac{1}{N_j^{act}/t_c} = \frac{1}{\left(N_j^{out} - N_j^{src}\right)/t_c} = \frac{t_c}{N_j^{out} - N_j^{src}} \quad (4)$$

where R_f is the one of the three parameters used to calculate the weight of each node, which represents the reciprocal of forward rate. F_r denotes the forward rate. N_j^{act} represents the number of packets forwarded by node j, as mentioned in Subsection 3.1.1. t_c is the time consumed to collect evidence. In other words, the N_j^{act} packets are observed in the period of t_c. N_j^{out} denotes the number of packets that "com out" of node j. N_j^{src} refers to the number of packets with node j as the source. Briefly, in calculating the parameter R_f, only the forwarded packets, rather than all the transmitted packets, are involved, i.e., the packets generated by node j are not considered into the calculation of R_f.

3.2 Calculation of Weight

We have introduced the calculation of three parameters, i.e., S_r, C_d and B_p. The calculation of weight for each node will then be presented by considering these three parameters. We use a powerful AHP methodology, a mathematical model, to compute the relative weights for all mobile nodes to select appropriate CHs in the network.

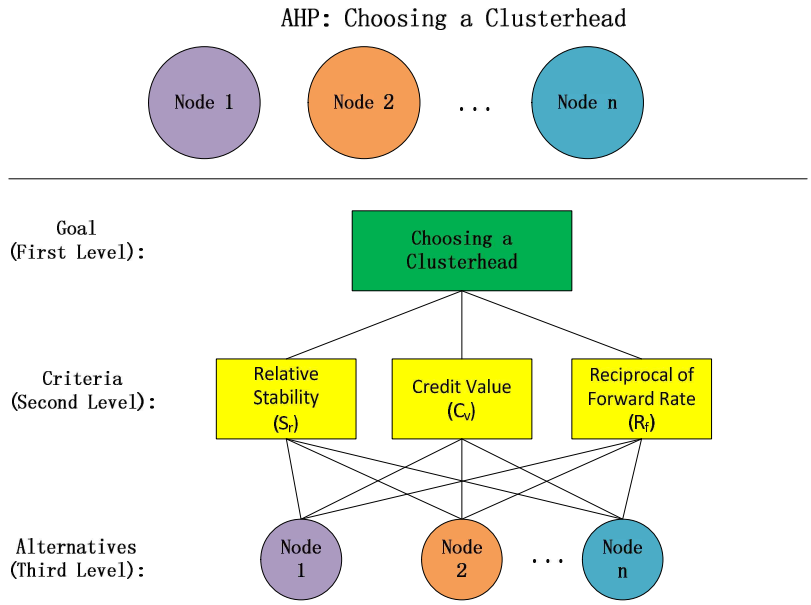

Fig. 4. Hierarchical structure

The election of a CH involves numerous metrics, including the aforementioned S_r, C_v and R_f, which need to be traded off. Hence, these metrics have to be measured, i.e., the measurements of these metrics must also be evaluated as to how well they serve the objectives of the decision of the CH maker. The judgments may be inconsistent, and the evaluation of inconsistency and improvement of the judgments, when possible, to obtain better consistency is a concern of the AHP. The derived priority scales are synthesized by multiplying them by the priority of their parent nodes and adding them for all such nodes. To make a decision on CH election in an organized way to generate priorities, we need to decompose the decision using the four steps [9] below:

Step 1: Define the problem. The decision problem in this paper involves the selection of a proper node by considering the corresponding weight values of the candidates nodes among the one-hop vicinity in the environment of ad hoc networks.

Step2: Build the decision hierarchy in such a way that the goal of the decision is on top so that the objectives are set from a broad perspective, followed by the intermediate levels (the criteria on which the subsequent elements depend) and the lowest level (which is usually a set of alternatives). Figure 4 presents the structuring of a problem as a hierarchy. The overall objective of choosing an appropriate CH is placed as the topmost goal of the hierarchy. The subsequent level representing the main criteria is called the secondary goal. The three secondary goals are S_r, C_v and R_f. Finally, the alternatives are placed at the bottom level of the hierarchy that is evaluated for the selection of the CH.

Table 2. Fundamental scale of absolute numbers

Level of importance	Definition	Explanation
1	Equal importance	Two activities contribute equally to the objective.
2	Weak or slight	Experience and judgment slightly favor one activity over another.
3	Moderate importance	
4	Moderate plus	Experience and judgment strongly favor one activity over another.
5	Strong importance	
6	Strong plus	An activity is favored very strongly over another; its dominance is demonstrated in practice.
7	Very strong	
8	Very, very strong	The evidence favoring one activity over another is of the highest possible order of affirmation.
9	Extreme importance	

Step 3: Construct a set of pairwise comparison matrices.

To make comparisons, we need a scale of numbers that indicates how many times more important or dominant one element is over another element with respect to the criterion or property with which they are compared. Table 2 exhibits the scale. The fundamental 1 to 9 scale is utilized to express the strength of preference based on intuition, experience and knowledge.

The criteria matrix A in Equation 5 gives the pair-wise comparison of three criteria toward the top-most goals. The reciprocal matrix is constructed through the pair-wise comparison of each criterion against another under the topmost goal. The values of the pair-wise comparison matrices are provided by answering the questions as to which is more preferred and by how much.

$$A = [a_j] = \begin{bmatrix} S_r \\ C_v \\ R_f \end{bmatrix} [S_r \quad C_v \quad R_f] = \begin{bmatrix} 1 & a_{S_r C_v} & a_{S_r R_f} \\ 1/a_{S_r C_v} & 1 & a_{C_v R_f} \\ 1/a_{S_r R_f} & 1/a_{C_v R_f} & 1 \end{bmatrix} \quad (5)$$

where a_j denotes the strength of preference of the i^{th} criteria over j^{th} criteria.

Through mean normalization of row vector, matrix A can be standardized to a normalized vector matrix A^{norm}, as shown in Equation 6.

$$A^{norm} = \left[\frac{a_j}{(\sum_{i=1}^{k} a_j)} \right] \quad (6)$$

where k is the number of criteria. In our scheme, three parameters are referred to as the criteria for evaluating the weight value of each node, so that k is equal to 3.

Through mean normalization of row vector, we can obtain the normalized vector W_i^T, as shown in Equation 7, which stands for the weight factor of each criterion.

$$W_i^T = [w_j] = \left[\frac{1}{k}\sum_{j=1}^{k}\left(\frac{a_j}{(\sum_{i=1}^{k} a_j)}\right)\right] \quad (7)$$

All pair-wise comparison matrices are checked for consistency. Considering that judgements are random, the matrices may be prone to judgments errors that can be detected by the Consistency Ratio (CR), which is defined as the ratio of the Consistency Index (CI) to the Random Index (RI). CI can be calculated using Equation 8 and is shown for a criteria matrix C as an instance. After the calculation of weight for each criterion, consistency should be considered.

$$CI = \frac{\lambda - n}{n-1} \quad (8)$$

where n denotes the number of elements to be compared in the criteria matrix A, and in this case, it is equal to 3. λ can be calculated using Equation 9.

$$\lambda = \frac{\sum_{i=1}^{n} \mu_i}{n} \quad (9)$$

where μ_i is the consistency vector which can be calculated using Equation 10.

$$\mu_i = \frac{\sum_{j=1}^{n} w_j a_j}{w_i} \quad (10)$$

where w_i is the weight factor of each criterion calculated by the aforementioned Equation 7.

Table 3. Random index

Exponent Number	1	2	3	4	5	6
RI	0	0	0.58	0.90	1.12	1.24

Finally we can obtain the CR, which is the ratio of CI to RI, as shown in Equation 11. RI is shown in Table 3. When $CR < 0.1$, the consistency of the matrix is high and acceptable, i.e., judgment errors are tolerable; otherwise, the pair-wise matrix undergoes certain adjustments until the it satisfies the consistency check.

$$CR = \frac{CI}{RI} \quad (11)$$

To explain the process of CH election clearly and intuitively, we show an example by assuming the weight of the second-level criteria. For example, if we consider the criteria for S_r as moderately more important than the criteria for C_v, we can assign "3" to $a_{S_r C_v}$. If we consider the criteria for S_r to be extremely more important than that for R_f, we can assign "9" to $a_{S_r R_f}$. If we consider the criteria for C_v to be strongly more important than that for R_f, we can assign "6" to $a_{C_d R_f}$. Hence, matrix A is shown in Equation 12:

$$A = [a_j] = \begin{bmatrix} S_r \\ C_v \\ R_f \end{bmatrix} [S_r \quad C_v \quad R_f] = \begin{bmatrix} a_{S_rS_r} & a_{S_rC_v} & a_{S_rR_f} \\ 1/a_{S_rC_v} & a_{C_vC_v} & a_{C_vR_f} \\ 1/a_{S_rR_f} & 1/a_{C_vR_f} & a_{R_fR_f} \end{bmatrix} =$$

$$\begin{bmatrix} 1 & a_{S_rC_v} & a_{S_rR_f} \\ 1/a_{S_rC_v} & 1 & a_{C_vR_f} \\ 1/a_{S_rR_f} & 1/a_{C_vR_f} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 3 & 9 \\ 1/3 & 1 & 6 \\ 1/9 & 1/6 & 1 \end{bmatrix} \quad (12)$$

Using Equation 6, matrix A can be standardized to a normalized A^{norm}, as shown in Equation 13.

$$A^{norm} = \begin{bmatrix} a^{norm}_{S_rS_r} & a^{norm}_{S_rC_v} & a^{norm}_{S_rR_f} \\ a^{norm}_{C_vS_r} & a^{norm}_{C_vC_v} & a^{norm}_{C_vR_f} \\ a^{norm}_{R_fS_r} & a^{norm}_{R_fC_v} & a^{norm}_{R_fR_f} \end{bmatrix} = \begin{bmatrix} 0.6923 & 0.7200 & 0.5625 \\ 0.2308 & 0.2400 & 0.3750 \\ 0.0769 & 0.0400 & 0.0625 \end{bmatrix} \quad (13)$$

where $a^{norm}_{S_rC_v} = \frac{a_{S_rC_v}}{a_{S_rC_v} + a_{C_vC_v} + 1/a_{C_vR_f}} = \frac{3}{3+1+1/6} = 0.7200$, for instance.

The weight of each criterion can then be calculated using Equation 7.

The weight of S_r: $W_{S_r} = \frac{a^{norm}_{S_rS_r} + a^{norm}_{S_rC_v} + a^{norm}_{S_rR_f}}{3} = \frac{(0.6923 + 0.7200 + 0.5625)}{3} = 0.6583$

The weight of C_v: $W_{C_v} = \frac{a^{norm}_{C_vS_r} + a^{norm}_{C_vC_v} + a^{norm}_{C_vR_f}}{3} = \frac{(0.2308 + 0.2400 + 0.3750)}{3} = 0.2819$

The weight of R_f: $W_{B_p} = \frac{a^{norm}_{R_fS_r} + a^{norm}_{R_fC_v} + a^{norm}_{R_fR_f}}{3} = \frac{(0.0769 + 0.0400 + 0.0625)}{3} = 0.0598$

Hence, using Equation 5, the vector of weight of each criterion W_i^T can be calculated as shown in Equation 14.

$$W_i^T = \frac{1}{k}\sum_{j=1}^{k}\left(\frac{a_j}{(\sum_{i=1}^{k} a_j)}\right) = [W_{S_r} \quad W_{C_v} \quad W_{R_f}] = [0.6583 \quad 0.2819 \quad 0.0598] \quad (14)$$

We then consider the consistency problem. The CI should be calculated to measure the weight of each criterion above, i.e., whether W_{S_r}, W_{C_v} and W_{R_f} are consistent. Using Equation 10, the consistency vector μ_i can be calculated, as shown in Equation 15.

$$\mu_i = \begin{bmatrix} (W_{S_r} \cdot a_{S_rS_r} + W_{C_v} \cdot a_{S_rC_v} + W_{R_f} \cdot a_{S_rR_f})/W_{S_r} \\ (W_{S_r} \cdot 1/a_{S_rC_v} + W_{C_d} \cdot a_{C_vC_v} + W_{R_f} \cdot a_{C_vR_f})/W_{C_v} \\ (W_{S_r} \cdot 1/a_{S_rR_f} + W_{C_v} \cdot 1/a_{C_vR_f} + W_{R_f} \cdot a_{R_fR_f})/W_{R_f} \end{bmatrix} =$$

$$\begin{bmatrix} (0.6853 \cdot 1 + 0.2819 \cdot 3 + 0.0598 \cdot 9)/0.6853 \\ (0.6853 \cdot 1/3 + 0.2819 \cdot 1 + 0.0598 \cdot 6)/0.2819 \\ (0.6853 \cdot 1/9 + 0.2819 \cdot 1/6 + 0.0598 \cdot 1)/0.0598 \end{bmatrix} = \begin{bmatrix} 3.1025 \\ 3.0512 \\ 3.0086 \end{bmatrix} \quad (15)$$

Using Equation 9, we can calculate the value of λ as shown in Equation 16.

$$\lambda = \frac{\sum_{i=1}^{n} \mu_i}{n} = \frac{3.1025 + 3.0512 + 3.0086}{3} = 3.0541 \quad (16)$$

Using Equation 8, we derive the value of CI from Equation 17.

$$CI = \frac{\lambda - n}{n-1} = \frac{3.0541 - 3}{3-1} = 0.0270 \quad (17)$$

From Table 3, we can obtain the RI based on the value of n which is 3. Finally, we can calculate the value of CR. As shown in Equation 18, CR is less than 0.1; hence, the consistency is acceptable.

$$CR = \frac{CI}{RI} = \frac{0.0270}{0.58} = 0.0466 < 0.1 \quad (18)$$

where RI is equal to 0.58, as obtained from Table 3, given that n is equal to 3.

Step 4: Use the priorities obtained from the comparisons to weigh the priorities in the level immediately below. Do this for every element. Then for each element in the level below add its weighed values and obtain its overall or global priority. Continue this process of weighing and adding until the final priorities of the alternatives at the bottom most level are obtained.

After calculating the weight of each criterion, we should calculate the weight of each node using the same process. In our CH election process, three criteria are considered, i.e., S_r, C_v and R_f. The nodes within the vicinity should be compared based on every perspective of each criterion to obtain matrices A_{S_r}, A_{C_v}, and A_{R_f}, which are the local weights factor of the corresponding parameters, as shown in Equation 19, 20, and 21. To provide a clear and intuitive explanation, we assume that only three nodes are present in the vicinity, i.e., node 1 (n_1), node 2 (n_2), and node 3 (n_3). A_{S_r} stands for the pairwise comparisons of nodes in the vicinity according to the criterion of S_r. A_{C_v} stands for the pairwise comparisons of nodes in the vicinity according to the criterion of C_v. A_{R_f} stands for the pairwise comparisons of nodes in the vicinity according to the criterion of R_f.

$$A_{S_r} = \left[a_j^{S_r}\right] = \begin{bmatrix} a_{n_1 n_1}^{S_r} & a_{n_1 n_2}^{S_r} & a_{n_1 n_3}^{S_r} \\ 1/a_{n_1 n_2}^{S_r} & a_{n_2 n_2}^{S_r} & a_{n_2 n_3}^{S_r} \\ 1/a_{n_1 n_3}^{S_r} & 1/a_{n_2 n_3}^{S_r} & a_{n_3 n_3}^{S_r} \end{bmatrix} \quad (19)$$

$$A_{C_v} = \left[a_j^{C_v}\right] = \begin{bmatrix} a_{n_1n_1}^{C_v} & a_{n_1n_2}^{C_v} & a_{n_1n_3}^{C_v} \\ 1/a_{n_1n_2}^{C_v} & a_{n_2n_2}^{C_v} & a_{n_2n_3}^{C_v} \\ 1/a_{n_1n_3}^{C_v} & 1/a_{n_2n_3}^{C_v} & a_{n_3n_3}^{C_v} \end{bmatrix} \quad (20)$$

$$A_{B_p} = \left[a_j^{R_f}\right] = \begin{bmatrix} a_{n_1n_1}^{R_f} & a_{n_1n_2}^{R_f} & a_{n_1n_3}^{R_f} \\ 1/a_{n_1n_2}^{R_f} & a_{n_2n_2}^{R_f} & a_{n_2n_3}^{R_f} \\ 1/a_{n_1n_3}^{R_f} & 1/a_{n_2n_3}^{R_f} & a_{n_3n_3}^{R_f} \end{bmatrix} \quad (21)$$

We first calculate A_{S_r}, which is based on the criterion of S_r. For example, if we consider that the relative stability value of n_2 is slightly larger than that of n_1, we can assign "1/2" to $a_{n_1n_2}^{S_r}$. If we consider that the relative stability value of n_3 is very strongly larger than that of n_1, we can assign "1/8" to $a_{n_1n_3}^{S_r}$. If we consider the relative stability value of n_3 is strongly larger than that of n_2, we can assign "1/5" to $a_{n_2n_3}^{S_r}$. Hence, matrix A_{S_r} is formed in Equation 22.

$$A_{S_r} = \left[a_{ij}^{S_r}\right] = \begin{bmatrix} a_{n_1n_1}^{S_r} & a_{n_1n_2}^{S_r} & a_{n_1n_3}^{S_r} \\ 1/a_{n_1n_2}^{S_r} & a_{n_2n_2}^{S_r} & a_{n_2n_3}^{S_r} \\ 1/a_{n_1n_3}^{S_r} & 1/a_{n_2n_3}^{S_r} & a_{n_3n_3}^{S_r} \end{bmatrix} = \begin{bmatrix} 1 & 1/2 & 1/8 \\ 2 & 1 & 1/5 \\ 8 & 5 & 1 \end{bmatrix} \quad (22)$$

From the similarity assumption, we can obtain A_{C_v} and A_{R_f}, as shown in Equations 23 and 24.

$$A_{C_v} = \left[a_{ij}^{C_v}\right] == \begin{bmatrix} 1 & 1 & 6 \\ 1 & 1 & 3 \\ 1/6 & 1/3 & 1 \end{bmatrix} \quad (23)$$

$$A_{R_f} = \left[a_{ij}^{R_f}\right] == \begin{bmatrix} 1 & 1/8 & 1/3 \\ 8 & 1 & 3 \\ 3 & 1/3 & 1 \end{bmatrix} \quad (24)$$

Using the same process of calculation for W_i^T, we can obtain the weight factor of every node on the corresponding criterion, as shown in Equation 25.

$$\propto = [\propto_{ij}] = \begin{bmatrix} \propto_{n_1}^{S_r} & \propto_{n_2}^{S_r} & \propto_{n_3}^{S_r} \\ \propto_{n_1}^{C_v} & \propto_{n_2}^{C_v} & \propto_{n_3}^{C_v} \\ \propto_{n_1}^{R_f} & \propto_{n_2}^{R_f} & \propto_{n_3}^{R_f} \end{bmatrix} = \begin{bmatrix} 0.0874 & 0.1622 & 0.7504 \\ 0.4967 & 0.3967 & 0.1066 \\ 0.082 & 0.6816 & 0.2364 \end{bmatrix} \quad (25)$$

where $\propto_{n_1}^{S_r}$ stands for the instance when only the criterion of S_r is considered. The weight of node 1 is 0.0874.

$$W_i = \sum_{j=1}^{n} w_j \cdot \propto_{ij} \quad (26)$$

The global weight of a mobile node is obtained by multiplying its local weight by its corresponding parent weights. From Equation 26, we can derive the global weight vector, as shown in Equation 27.

$$W_i = [W_{n_1} \quad W_{n_2} \quad W_{n_3}] = [0.2025 \quad 0.2594 \quad 0.5382] \quad (27)$$

where W_{n_1} is the weight of node 1 after considering all criteria, $W_{n_1} = W_{S_r} \cdot \propto_{n_1}^{S_r} + W_{C_v} \cdot \propto_{n_1}^{C_v} + W_{R_f} \cdot \propto_{n_1}^{R_f} = 0.6583 \cdot 0.0874 + 0.2819 \cdot 0.4967 + 0.0598 \cdot 0.082 = 0.2025$.

The CH can then be elected because the global weight of each node in the vicinity has been determined. In our example, n_3 has the largest weight value, i.e., 0.5382, as shown in Equation 27. Therefore, n_3 is elected as the CH for the nodes in the vicinity.

3.3 Detecting MAC Layer Misbehavior

In this section, we use two phases to detect MAC layer misbehavior, explained in the following two subsections.

3.3.1 Phase 1: Assignment of Backoff

Using the aforementioned clustering algorithm, clusters can be formed, and a CH can be identified the in each cluster. The process for assigning the backoff is shown below.

1) The originator broadcasts the RTS to request the channel, which is the same as the specification in the 802.11 DCF. The difference is that the RTS piggybacks the ID of the CH. All the member nodes in the cluster, including the CH, will receive the RTS;

2) Upon receiving the RTS from the originator node, the CH replies to the CTS, piggybacking the backoff value chosen from the range [0, CW_{min}]. CW_{min} is the minimum contention window value used in IEEE 802.11;

3) Considering that the CTS might be lost for a number of reasons (e.g. bad channel condition), the CH sets a timeout value and observes the behavior of the originator during the period. If the originator does not send any packet during the period, timeout occurs. The CH will resend the CTS, with the backoff value selected from the range [0, $2CW_{min}$];

4) The originator receives the CTS message and verifies whether the source of the CTS is the CH, in which case, the originator extracts the backoff value. The originator may then use the value as its initial backoff for the following transmission.

The aforementioned procedure is intended for the assignment of backoff, i.e., the CH is requested to set the backoff value for the originator rather than the originator choosing the backoff value itself. The next phase in the following subsection is intended to allow the CH to continue monitoring the originator.

3.3.2 Phase 2: Detection of MAC Layer Misbehavior

Aside from assigning backoff values to the originator, the CH also continues monitoring whether the originator complies with the backoff offered by the CH. The implementation of the monitoring can be achieved through a watchdog mechanism. The CH can determine whether the originator is a normal node, a MAC layer misbehaving node, or a selfish node that does not actively participate in the network by comparing the expected backoff value selected by the CH and the actual backoff value used by the originator.

Table 4. Parameters for detecting MAC layer misbehavior mechanism

Parameter	Explanation
b_{exp}	Expected backoff size set by the CH
b_{act}	Actual backoff size s used by the originator
b_{diff}	Difference between b_{exp} and b_{act}
α, β	Threshold (α, $\beta > 0$)
n	Transmission times of the originator
i	i^{th} time transmission of the originator

The need for this phase is explained as follows: Although in our scheme, the CH node is requested to set the backoff to the originator, the originator does not necessarily have to use the backoff. This backoff is only a reference value. The size of the actual backoff should be based on the channel condition at a given moment. For example, if the channel condition is good, the originator could perform the routing or transmission actions faster, i.e., choose a smaller backoff value. However, if the channel condition is bad, the originator could choose a larger backoff value by itself to reduce the collsion.

The key idea of this phase is to monitor the behavior of the originator by comparing the expected backoff (b_{exp}) and the actual backoff (b_{act}). By comparing the two backoff values, the originator may be judged as an attacker that consumes bandwidth unfairly, a selfish node that does not actively participate, or a normal node.

The difference between expected backoff and actual backoff can be calculated using Equations 28, 29, 30 and 31. The Parameters are explaned in Table 4.

$$b_{diff_i} = b_{exp_i} - b_{act_i} \tag{28}$$

$$\frac{\sum_{i=1}^{n} b_{diff_i}}{\sum_{i=1}^{n} b_{exp_i}} > \alpha \tag{29}$$

If Equation 29 is satisfied, the node is a MAC layer misbehaving node, which consumes bandwidth unfairly and causes the medium to appear busy to other normal nodes.

$$\frac{\sum_{i=1}^{n} b_{diff_i}}{\sum_{i=1}^{n} b_{exp_i}} < -\beta \tag{30}$$

If Equation 30 is saticfied, the node is a selfish node, which does not actively participate in the networks to save its resources (e.g., battery power).

$$-\beta \leq \frac{\sum_{i=1}^{n} b_{diff_i}}{\sum_{i=1}^{n} b_{exp_i}} \leq \alpha \tag{31}$$

If Equation 31 is satisfied, the node is a normal node.

The reference factors of both α and β depend on the channel condition. If the channel condition is good, α is set as a larger value (e.g., 0.5), and β is set as a smaller value (e.g., 0.2). The reason is that when the channel condition is good, the originator may choose a smaller backoff to decrease the transmission delay. However, if the channel condition is bad, α is set as a smaller value (e.g. 0.2), and β is set as a larger value (e.g., 0.5). The reason is that when the channel condition is bad, the originator may choose a larger backoff to decrease the collision.

4. Discussion and Future Work

In this section, we discuss the advantages and disadvantages of our scheme compared with other methods. We then address the direction of our future work.

As mentioned in Section 2, one of the major limitations of using game theory to prevent misbehavior is that game theory protocols assume that all nodes are selfish, which differs from the reality of ad hoc networks. On the contrary, our scheme is based on the assumption that the majority of nodes exhibit good-behavior, and the minority comprises misbehaving nodes. Under this assumption, we focus on misbehaving nodes. Another issue with some protocols based on game theory is that they assume that all nodes are within the wireless range, which is not satisfied in practical ad hoc networks. In our scheme, the CHs are in different clusters that provide security for the corresponding originators; thus, scalability can be guaranteed. An additional benefit of our scheme is that considering the distribution of CHs in the whole network, resource consumption is divided and shared among CHs, an arrangement that performs better than that in central-based networks.

The key issue in [5] is that the receiver assigns the backoff values to the originator, which means that the receiver must be a trustworthy node. However, in ad hoc networks each node has an equalized security status. The trustworthiness of the receiver cannot be guaranteed. Thus, our scheme utilizes and expands this idea by allowing the CH, rather than the receiver, to assign the backoff values to the originator. The benefit is that the

node that assigns the backoff values is trustworthy and the assigned backoff value is dependable.

The main drawback of our previous paper [10] is the lack of scalability in large scale ad hoc networks. As mentioned in Section 2, this drawback is mainly caused by the reactive election of LMT nodes. However, in this paper, we utilize the benefit of cluster hierarchy to elect CHs that set the backoff value to the originators, a method that works well even for large-scale ad hoc networks. Moreover, we introduce another parameter, i.e., the reciprocal of forward rate, to measure the node's packet forwarding behavior and rate of battery power consumption. In our previous work [10], the trust value of each node is simply the summary of two weighted parameters. The weight factor for each parameter is assigned by the authors subjectively and without any justification of rationality. However, in this paper, we use the AHP model to evaluate the weight value of each node, and then supply the proof for justifying the rationality of the weight factor for each parameter.

However, our scheme has an overhead to calculate the weight value.

1) For our clustering algorithm, when no transmission exists at initial time, the number of packets actually forwarded and the number of packets expected to be forwarded cannot be determined. Thus, the credit value and the reciprocal of forward rate cannot be evaluated at initial time. The values can be evaluated only after a period of transmission, which causes delay in the evaluation of the weight value of each node.

2) When a node moves to a new cluster, the node becomes a stranger to its neighbors. Hence, the node's weight value must be recalculated, and the problem mentioned above may arise again. If the node moves frequently, this problem may become more serious. However, the stability value is a variable of the weight value function in our scheme. If the node moves frequently, its stability value will be small as shown in Equation 1. Accordingly, the weight value of the node will be small, which means that the node will not become the CH with the largest weight value in the neighborhood.

Our future research directions for the enhancement of our scheme are as follows:

1) We will consider more parameters into the calculation of the weight of each node, including the cumulative time during which the CH had been in the last cluster. Cumulative time implies node stability, which increases the stability of cluster. The purpose of including more parameters is to enhance clusters stability and to reduce the frequencies of CH reelection. We will measure more parameters and then select the more important and appropriate ones to strike a balance between the appropriateness of the CH and the efficiency of weight calculation and exchange.

2) We will enhance the clustering algorithm to reduce the delay in determining the CH for each cluster.

3) We will address the reaction mechanism of CHs after detecting the misbehavior in the MAC layer.

4) In this paper, we only consider about a simple attacker which has no knowledge

about our security system. In our future work, we plan to mitigate a smart attacker that knows how the security system works and that performs smarter attacks.

5. Conclusion

This paper presents an initial work on the detection of misbehavior in the MAC layer that is caused by the misuse of backoff values in the 802.11 DCF in an ad hoc network. An avoidance mechanism is used to detect this type of attack. In our scheme, the CH is elected based on the AHP mathematical model to assign backoff values to the originator. The CH monitors the actual backoff used by the originator to determine whether the originator is a misbehaving node.

References

[1] Kannhavong B., Nakayama H., Nemoto Y., Kato N., and Jamalipour A., "A survey of routing attacks in mobile ad hoc networks," *Wireless Communications*, vol.14, no.5, pp. 85-91, 2007. Article (CrossRef Link).

[2] D. Wang, M. Hu, and H. Zhi, "A survey of secure routing in ad hoc networks," in *Proc. of 9th International Conference on Web-Age Information Management*, pp.482-486, Jul.2008. Article (CrossRef Link).

[3] K. Hong, S. Lee, K. Kim, and Y. Kim, "Channel condition based contention window adaptation in IEEE 802.11 WLANs," *IEEE Transactions on Communications*, vol.60, no.2, pp.469-478, 2012. Article (CrossRef Link).

[4] M. Cagalj, S. Ganeriwal, I. Aad, and J. P. Hubaux, "On cheating in CSMA/CA ad hoc networks," *EPFL Technical report*, No. IC/2004/27, 2004. http://www.mendeley.com/research/on-cheating-in-csmaca-ad-hoc-networks/.

[5] Kyasanur P. and Vaidya N, "Detection and handling of MAC layer misbehavior in wireless networks," in *Proc. of International Conference on Dependable Systems and Networks*, pp.173-82, Jun.2003. Article (CrossRef Link).

[6] J. Liu, P. Geng, Y. Qiu, and G. Feng, "A secure routing mechanism in AODV for Ad Hoc networks," in *Proc. of International Symposium on Intelligent Signal Processing and Communication Systems*, pp.435-438, Nov.2007. Article (CrossRef Link).

[7] András Faragó, "Scalable analysis and design of ad hoc networks via random graph theory," in *Proc. of 6th international workshop on Discrete algorithms and methods for mobile computing and communications*, pp.43-50, Sep.2002. Article (CrossRef Link).

[8] Tsumoto S., and Hirano S., "Visualization of rule's similarity using multidimensional scaling," in *Proc. of 3rd IEEE International Conference on Data Mining*, pp.339-346, Nov.2003. Article (CrossRef Link).

[9] Thomas L. Saaty, "Decision making with the analytic hierarchy process," *International Journal of Services Sciences*, vol.1, no.1, pp.83-98, 2008. Article (CrossRef Link).

[10] Fei Shi, Jaejong Baek, Jooseok Song, and Weijie Liu, "A novel scheme to prevent MAC layer misbehavior in IEEE 802.11 ad hoc networks," *Journal of Telecommunication Systems*, 2011. Article (CrossRef Link).

[11] Dilip Kumar, Trilok C. Aseri and R.B. Patel, "Multi-hop communication routing (MCR) protocol for heterogeneous wireless sensor networks," *International Journal of Information Technology, Communications and Convergence*, vol.1, no.2, pp.130-145, 2011. Article (CrossRef Link).

Fei Shi received the B.S. degree in computer science and M.S. degree in computer system from the Northeastern University, Shenyang, China, in 2004, and 2008, respectively. He is currently a Ph.D. candidate in computer science at Yonsei University, Seoul, Korea. His research interests include wireless communications, mobile ad hoc networks, vehicular ad hoc networks and information security.

JooSeok Song received the B.S. degree in Electrical Engineering from Seoul National University, Korea, in 1976, and the M.S. degree in Electrical Engineering from Korea Advanced Institute of Science and Technology, Korea, in 1979. In 1988, he received the Ph.D. degree in Computer Science from University of California at Berkeley. From 1988 to 1989, he was an Assistant Professor at the Naval Postgraduate School, Monterey, CA. He was the president of Korea Institute of Information Security and Cryptology in 2006. He is currently a Professor of Computer Science at Yonsei University, Seoul. His research interests include cryptography and network security.