International Journal of Security and Its Applications
Vol.7, No.6 (2013), pp.1-10
http://dx.doi.org/10.14257/ijsia.2013.7.6.01

An Encryption Based Black Hole Detection Mechanism in Mobile Ad Hoc Networks

Firoz Ahmed[1] and Hoon Oh[2]

[1]*Department of Information and Communication Engineering*
University of Rajshahi, Rajshahi-6205, Bangladesh
[2]*School of Electrical Engineering*
University of Ulsan, P.O. Box 18, Ulsan 680-749, South Korea
[1]*jewelraaz@yahoo.com*, [2]*hoonoh@ulsan.ac.kr*

Abstract

A black hole attack is one of the most serious attacks in mobile ad hoc networks. A malicious node can act as if it has a valid route to a destination and then respond with a false Route Reply (RREP) message to the source, when it receives a Route Request (RREQ). Then the malicious node absorbs data packets destined for the destination. We propose an Encrypted Verification Method (EVM) that effectively detects a black hole attack. It takes two steps. First, every node examines its neighbors by inspecting their data transmission behaviors. Second, a detection node that receives an RREP from the suspicious node sends an encrypted verification message directly to the destination along the path included in the RREP for verification. The approach not only pins down the black hole nodes, but also reduces control overhead significantly. We prove by resorting to simulation that the EVM is highly dependable against the black hole attack.

Keywords: *Black hole attack, encryption, decryption, verification, MANET*

1. Introduction

Mobile ad hoc networks (MANETs) are formed autonomously by a number of mobile nodes (MNs) without the help of a centralized management entity. Every MN can act as a router that forwards data packets to another MN along a pre-established path. Since an MN can join or leave the network without permission from a management entity, MANETs are vulnerable to various kinds of attacks such as a black hole attack [1], worm hole attack [2], gray hole attack [3] and so on. We address the black hole attack problem when AODV [4] is used for routing in MANETs.

In AODV, a malicious node may respond with an RREP that includes a higher sequence number so that it can absorb data packets destined for the destination. This type of attack is called a *black hole attack*. There are two types of black hole attacks: A single black hole attack and a colluding black hole attack. The former is made by a single node while the latter is made by two or more nodes that are collaborating to deceive the other nodes more effectively.

The single black hole attack has been tackled in various ways. Some focused on verifying the correctness of the obtained path through the downstream node of the RREP initiator [1, 5]. These approaches use an additional flooding message in every issue of the RREP. On the other hand, a watchdog mechanism has been proposed where a node watches the misbehavior of its downstream node [6]. However, it may not work appropriately if two black hole nodes collude to deceive the watchdog. Meanwhile, only

ISSN: 1738-9976 IJSIA

a few methods have been proposed to tackle the colluding black hole attack. In SNV [7], every RREP initiator sends a message to destination to ask for the destination to report its current sequence number to the source. However, it produces high control overhead and tends to make a false decision in detecting a malicious node, especially in the relatively high mobility networks.

The approaches discussed above suffer from high overhead by using flooding or additional messages as well as the failure to address a colluding attack. We propose an *encrypted verification method* (EVM) that uses an encrypted verification message to resolve the above problems effectively. It consists of two steps: The identification of a suspicious node and the verification of the suspicious node using an encrypted verification message. Thus, the malicious behaviors of a node on the path such as the fabrication, the dropping or the absorption of a message can be detected effectively. Simulation results show that the EVM can reduce control overhead and increase the detection rate considerably compared to the SNV.

The paper is organized as follows: The network model and problem identification in Section 2, the formal description of the proposed method in Section 3, performance evaluation in Section 4, and then concluding remarks in Section 5.

2. Background

2.1 Network Model

The network consists of a number of mobile nodes with a limited transmission range where a mobile node can join or leave the network freely. Every node can act as a router and can communicate with any node in the same network directly or via multiple wireless hops. A number of black hole nodes with a malicious intention can intrude the network. A routing protocol is used to establish a path between any two parties that want to communicate. The routing protocol used in this model is the AODV protocol.

2.2 Problem Identification

The watchdog mechanism [6] does not work effectively if multiple malicious nodes collaborate in order to deceive the watchdog such that one malicious node forwards the receiving packet to another malicious one on purpose. Another recently proposed method is the sequence number verification (SNV) method [7] in which source always verifies the sequence number contained in the RREP by receiving a current sequence number from the destination. However, this method has some shortcomings since the source that is multi-hop away from the RREP initiator has to make a decision, as follows:

- It can make a false decision such that either a normal node is determined to be malicious or a malicious node is to be normal; and
- It incurs high network overhead because the source that is multi-hop away from RREP-initiator always verifies the correctness of the destination sequence number and a node that detects the anomaly of a node always floods the network with an alarm message which is supposed to be delivered to a source.

For convenience of explanation, let us use some terminologies cited from the paper [10]. When a detection method is used to determine whether a node is malicious or not, it produces one of the four decisions TN (True Negative), FP (False Positive), FN (False Negative) and TP (True Positive).

- TN: The detection algorithm determines a normal node to be normal, thereby producing no alarm message
- FP: The detection algorithm determines a normal node to be malicious, thereby producing an alarm message
- FN: The detection algorithm determines a malicious node to be normal, thereby producing no alarm message
- TP: The detection algorithm determines a malicious node to be malicious, thereby producing an alarm message

Referring to Figure 1, the following three cases explain the problems of the SNV protocol.

Case 1: It can make an FP decision frequently in a network of relatively high node mobility:

A. A node may determine a normal RREP initiator to be malicious if it is disconnected from the RREP initiator immediately after it has received RREP, resulting in the initiation of an alarm message erroneously (see Figure 1-(a)).

B. A node may determine a normal RREP initiator to be malicious if the RREP initiator fails to send SREQ to its downstream node due to link breakage as in Figure 1-(b).

Case 2: Even though a node makes a TP decision, the alarm message can be lost easily by collision or link breakage since it has to go through wireless multiple hops. Thus, the source may not notice the existence of a malicious node. So, this TP decision will turn to an FN eventually (see Figure 1-(c)).

Case 3: Consider the colluding attack as illustrated in Figure 1-(d), If there does not exist a third node that connects commonly to the two colluding malicious nodes, the modification of sequence number cannot be detected, leading to an FN decision.

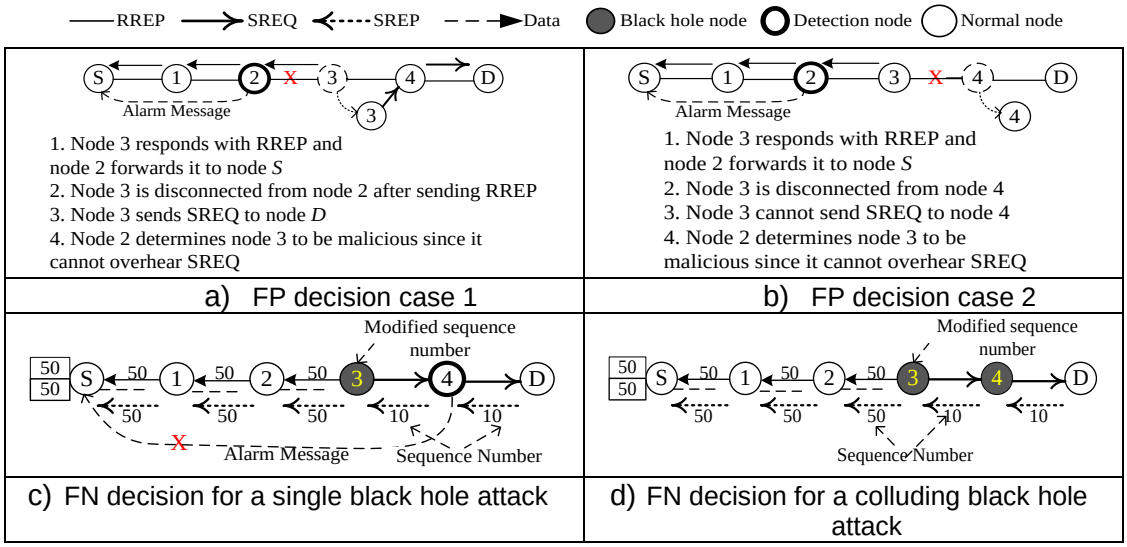


Figure 1. Shortcomings of the SNV Approach

Case 1-A and 1-B can be resolved if the upstream node of an RREP initiator initiates a verification message and if the RREP initiator sends error message to its immediate upstream node as a *detection node*. In Case 2, if the upstream node of a RREP initiator instead of source node is a verifier, it can be resolved easily. In Case 3, if we can prevent a node from modifying the test message and if the detection does not receive the test message within the estimated time it would not take any action such as forwarding RREP towards the source. In this case, the source will always try to consider other RREPs that it receives in order to establish a path.

Taking into consideration the problem-solving ways learnt from these observations, propose a new method to detect a black hole node.

3. Encrypted Verification Method (EVM)

3.1 Identification of Suspicious Node

Each node collects data necessary to identify a suspicious node by overhearing the packets that its neighbors transmit and maintains a data collection table (*DCT*) with those data as follows.

$DCT_i = (j, From_j, Through_j, Suspecious_j)$, $j \in i.N$, where

- $i.N$ is a collection of node i's neighbors;
- $From_j$ indicates whether or not node i has received a packet from node j ever;
- $Through_j$ indicates whether or not node i has routed a packet via node j ever; and
- $Suspicious_j$ indicates whether or not node j is suspicious based on the combination of $From_j$ and $Through_j$ fields.

The values of $From_j$, $Through_j$, and $Suspicious_j$ are given true (1), false (0), non-decidable (x).

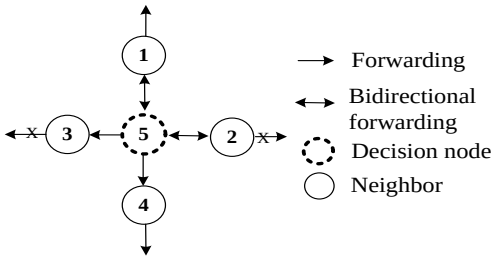


Figure 2. An Example Topology to Define *DCT*

Table 1. An Example of DCT_5

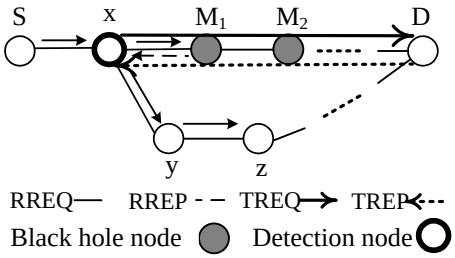
j	$From_j$	$Trough_j$	$Suspicious_j$
1	1	1	0
2	1	0	0
3	0	0	1
4	0	1	x

Take a look at Figure 2 and Table 1. Node 5 observes the data forwarding behaviors of its neighbors and records them in its data collection table, DCT_5. Node 5 has received data packet from nodes 1 and 2 ($From_1 = From_2 = 1$). Thus it determines that both are reliable ($Suspecious_1 = Suspecious_2 = 0$). However, node 3 did not send data to anyone, including node 5. Thus it is determined to be suspicious ($Suspecious_3 = 1$). As for node 4, node 5 cannot know whether node 4 has forwarded to a reliable node or a malicious node, and thus determines node 4 to be non-decidable ($Suspecious_4 = x$).

For the non-decidable node, we need further observation. It may be reasonable to assume that multiple different paths can go through the non-decidable node. A decision node can count the number of different downstream nodes to which the non-decidable node forward a data packet by using the watchdog mechanism. If the number is over some threshold, it can determine the non-decidable node to be reliable.

3.2 Verification Process

If a source or an intermediate node receives an RREP from a reliable node, it takes exactly the same process as AODV. That is, the source starts sending data packets while the intermediate node forwards the RREP to the source. A node that receives the RREP from a suspicious node, initiates a verification process to check if the suspicious node is a black hole.



1. *S* broadcasts RREQ to explore a path
2. M_1 responds with the fake RREP
3. If x determines M_1 to be suspicious, it sends the encrypted TREQ along the path specified in RREP
4. *D* receives the TREQ and responds with the encrypted TREP
5. *x* compares the sequence number in RREP and TREP to verify M_1

Figure 3. Verification Process

The node (detection node) extracts destination sequence number (dsn_1) from the RREP and stores it in its cache. It then generates a *Test Request message*, TREQ = (*detection node address, destination node address, timestamp*) where the *timestamp* indicates a current time. The detection node encrypts TREQ using public key cryptosystem and sends it along the path specified in the RREP towards the destination. A node that receives the TREQ relays it to the next node. Upon receiving the TREQ, the destination node decrypts the message and creates a *Test Reply message*, TREP = (*detection node address, destination node address, timestamp*, dsn_2) where dsn_2 indicates current destination sequence number. The destination encrypts the TREP and sends it along the reverse path to the detection node. Upon receiving the TREP, the detection node decrypts it. If $dsn_1 >> dsn_2$, the detection node judges the suspicious node is black hole and drops the RREP. If the suspicious node is determined to be a reliable node, the detection node starts sending data packets to the destination if the detection node is source; otherwise, it forwards the RREP towards the source node. If detection node does not receive TREP until the timer expires, it refuses to forward the RREP to its upstream node. Then, the source will consider other paths contained in some other RREPs that it has received.

Figure 3 illustrates a verification process. When a suspicious node M_1 responds with RREP, detection node *x* sends the encrypted TREQ to destination *D* along the path specified in RREP. Upon receiving the TREQ, node *D* responds with the encrypted TREP along the reverse path to the detection node *x*. M_1 and M_2 cannot alter the contents of the encrypted RREP. Detection node *x* judges whether M_1 is reliable or not by comparing dsn_1 and dsn_2.

3.2.1 Encryption and Decryption: We assume that every node has a pair of public and private keys that are used in the RSA public key cryptosystem [8]. The public key of each node can be distributed once when it joins to a considered network. The newly joined node can get the public keys of the other nodes in the network from one of its neighbors. The notations used in the cryptosystem are summarized in Table 2.

Table 2. Notations

Notations	Meaning
K_s+	Public key of node *s*
K_s-	Private key of node *s*
IP_s	IP address of node *s*
$[d]K_s$+	Message *d* encrypted with node *s*'s public key, K_s+
$[d]K_s$-	Message *d* signed with node *s*'s private key, K_s-
$s \rightarrow M[d]$	Node *s* sends message *d* to node *M*

If a node receives an RREP from a suspicious node, it (or detection node) extracts destination sequence number and stores it in its cache. The detection node creates TREQ and encrypts the detection node address with the public key of the destination node. Now, the detection node forwards the TREQ along the forward path towards the destination after it signs the whole TREQ with its own private key. The receiving node validates and removes the previous node's signature in the TREQ and checks whether it is the destination or not by checking the destination IP address. If it is not the destination, it takes the same process again. The process continues until the TREQ reaches the intended destination.

Given a forward path = (N_1, N_2 … N_{l-1}, N_l) where N_1 and N_l represent detection node and destination node, respectively. The behavior of each node on the path can be described formally as follows.

$$N_i \rightarrow N_{i+1} : \lfloor\lfloor IP_{N_1} \rfloor K_{N_l}+, IP_{N_l}, timestamp \rfloor K_{N_i}-, 1 \le i \le l-1$$

Upon receiving TREQ, the destination node creates TREP message in which both the destination node address and the dsn_2 are encrypted with the public key of detection node. Now, the destination node forwards the TREP along the reverse path after it signs the whole TREP with its own private key. The receiving node validates and removes the previous node's signature in the TREP and checks whether it is the detection node or not by checking the IP address of the detection node. If it is not the detection node, it takes the same process again. The process continues until the TREP reaches the detection node. The behavior of each node can be described formally as follows.

$$N_{l-i} \rightarrow N_{l-i-1} : \lfloor\lfloor IP_{N_l}, dsn_2 \rfloor K_{N_1}+, IP_{N_1}, timestamp \rfloor K_{N_{l-i}}-, 1 \le i \le l-1$$

4. Performance Evaluation

Table 3. Simulation Parameters

Parameter	Value
Number of nodes	50
Terrain range	1000 * 1000 m^2
Maximum speeds	0, 5, 10, 15, 20, 25m/s
Simulation time	300 sec.
Number of sessions	15
Number of malicious nodes	1, 2, 3, 4, 5

Using the NS-2 [9], we compare EVM and SNV using Random Waypoint Model. The used simulation parameters are given in Table 3. The simulation for each scenario was performed five times and then the average value for each metric was presented. We use four metrics: Packet delivery rate (PDR), control overhead (CO), true positive rate

(TPR) and false positive rate (FPR). The TPR indicates the ratio of the number of correct decisions to all the decisions made for the testing of malicious nodes (= *TP / (TP + FN)*) and the FPR does the ratio of the number of wrong decisions to all the decisions made for the testing of normal nodes (= *FP / (FP + TN)*).

Figure 4 and Figure 5 show packet delivery rate and control overhead with varying number of black hole nodes. The packet delivery rates of all schemes perform well in case of no black hole node. A significant result is that the packet delivery rate of AODV dramatically drops from 88 percent to 21 percent in the presence of one black hole node and it becomes worse as the number of black hole nodes increase. EVM has control overhead lower than SNV since it can send an encrypted verification message directly to a destination which is protected from the modification of other nodes. EVM can detect a black hole more reliably for the same reason. The control overhead of AODV sharply decreases as the number of malicious nodes increases since a black hole node tends to hinder normal protocol operation such as message forwarding. EVM shows control overhead lower than SNV since it does not use flooding.

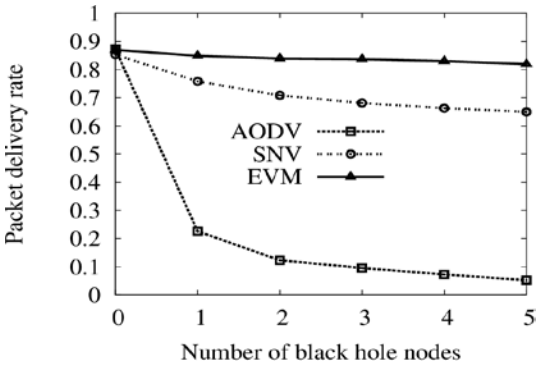


Figure 4. Packet Delivery Rate versus Number of Black Hole Nodes

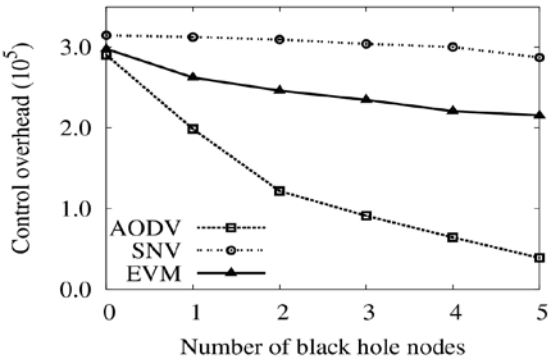


Figure 5. Control Overhead versus Number of Black Hole Nodes

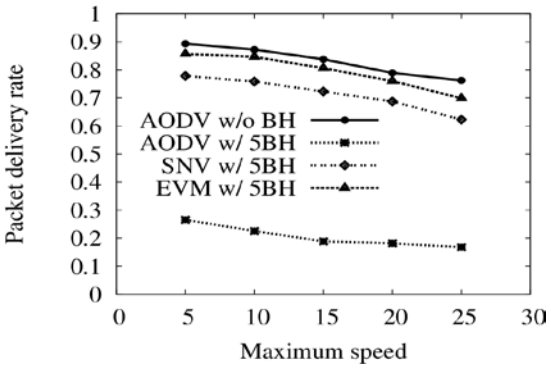


Figure 6. Packet Delivery Rate versus Maximum Speed

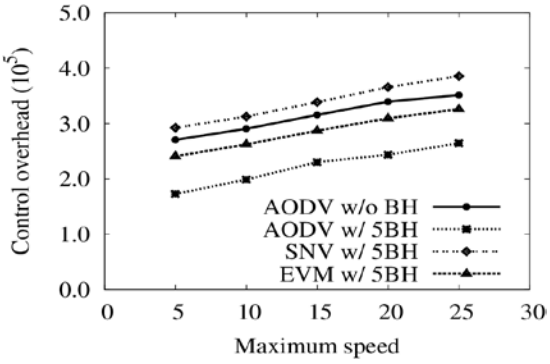


Figure 7. Control Overhead versus Maximum Speed

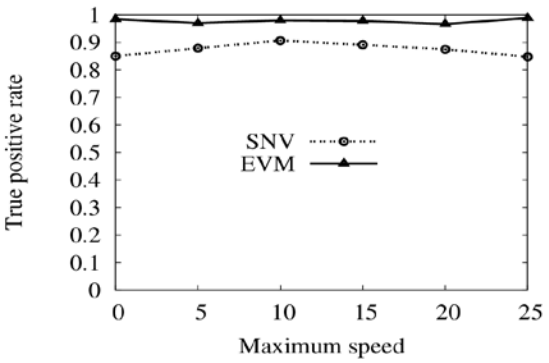


Figure 8. True Positive Rate versus Maximum Speed

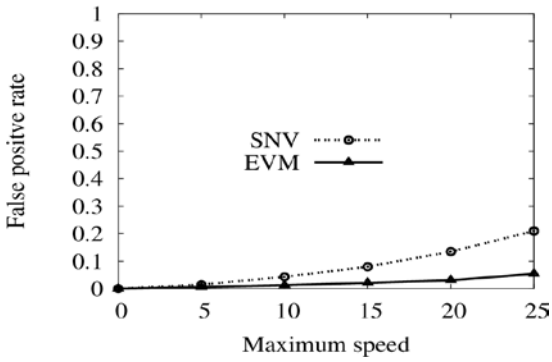


Figure 9. False Positive Rate versus Maximum Speed

In Figure 6 and Figure 7, we present packet delivery rate and control overhead as a function of a maximum speed while 10 percent of the total nodes are black hole node. As shown in Figure 6, the packet delivery rates of all schemes decrease when node mobility speed increases. On the other hand, the packet delivery rate of EVM is over 68 percent even with 30m/s, whereas that of AODV went down to 15 percent. The packet delivery rate of SNV is less than that of EVM because SNV cannot detect black hole attack effectively. Figure 7 shows that the control overhead of EVM with black hole node is lower than that of AODV without any black hole node. This is due to the fact that the EVM uses additional control messages, increasing control overhead; however, since control messages are unicast and conditional, the increase in overhead is not that serious. While the black hole node may limit the flooding of the RREQ partially, with the AODV in the network without a black hole node, the source floods the network with the RREQ without any hindrance. However, the control overhead of EVM is lower than that of SNV with the black hole node; the reason is as explained in Section 2.2.

Figure 8 and Figure 9 show the true and false positive rate with varying maximum speed while 10 percent of total nodes are black hole nodes. The true positive rate of SNV is lower than that of EVM because in SNV, only source node decides that a node is malicious. Therefore, if a certain node detects the anomaly of a node it floods the network with alarm message that is supposed to be delivered to the source. If the alarm message fails to reach the source, SNV cannot detect a black hole node. The false positive rate in both schemes increases when the nodes move more rapidly because links are broken frequently in a high mobility network. We also observe that the false positive rate of SNV is higher than that of EVM. This is due to the fact that in SNV, if the RREP initiator is disconnected from its immediate upstream node or it fails to send SREQ to its downstream node due to link breakage, the upstream node determines that the RREP initiator is a black hole node.

5. Conclusion

The proposed EVM method can pin down multiple black hole nodes effectively by employing an encryption mechanism. The verification process is initiated conditionally and the verification messages are delivered along the path including the malicious nodes in a unicast manner. This is possible since the messages or the sequence numbers contained in the messages cannot be modified by any malicious node. We show by simulation that the EVM not only reduces control overhead but also effectively identifies malicious nodes.

Acknowledgements

This work was supported by the development program of local science park funded by the ULSAN Metropolitan City and the MEST (Ministry of Education, Science and Technology).

References

[1] H. Deng, W. Li and D. P. Agarwal, "Routing Security in Wireless Ad Hoc Networks", IEEE Communications Magazine, vol. 40, no. 70, **(2002)**.
[2] Y.-C. Hu, A. Perrig and D. B. Johnson, "Packet Leashes: A Defense Against Wormhole Attacks in Wireless Ad Hoc Networks", Proceedings of the IEEE INFOCOM, New York, USA, **(2002)** June 23-27.
[3] D. Manikantan Shila, Yu Cheng and T. Anjali, "Mitigating Selective Forwarding Attacks with a Channel in WMNs", Journal of the IEEE Transaction on Wireless Communication, vol. 9, no. 5, **(2010)**.
[4] C. E. Perkins, E. M. Royer and S. Das, "Ad-hoc On demand Distance Vector (AODV) Routing Protocol", RFC, **(2003)** 3561.
[5] S. Lee, B. Han and M. Shin, "Robust Routing in Wireless Ad Hoc Networks", Proceedings of the 2002 International Conference on Parallel Processing Workshops, Vancouver, B.C., Canada, **(2000)** August 18-21.
[6] S. Marti, T. Giuli, K. Lai and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks", Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, Boston, USA, **(2000)** August 06-11.
[7] X. Yang Zhang, Y. Sekiya and Y. Wakahara, "Proposal of a Method to Detect Black Hole Attack in MANET", Proceeding of the 9th International Symposium on Autonomous Decentralized Systems, Athens, Greece, **(2009)** March 23-25.
[8] W. Stallings, "Cryptography and Network Security Principles and Practices", Prentice Hall Publishers, New Jersey, **(2005)**.
[9] NS-2, http://www.isi.edu/nsnam/ns/.
[10] A. Prathapani, L. Santhanam and D. P. Agrawal, "Detection of Black Hole Attack in a Wireless Mesh Network Using Intelligent Honeypot Agents", The Journal of Supercomputing (JoS), **(2011)**, pp. 1-28.

Authors

Firoz Ahmed is an associate professor of the department of Information and Communication Engineering (ICE) in the University of Rajshahi, Rajshahi, Bangladesh. He received M.Sc. degree from University of Madras, India in 2002 and Ph.D degree from University of Ulsan, South Korea, in 2012. His research interests include security in mobile ad hoc networks and wireless sensor network.

Hoon Oh is a professor of the School of Computer Engineering and Information Technology and a director of the Vehicle IT Convergence Technology Research Center in the University of Ulsan, Korea. He is a member of IEICW, ISCA, KICS, and ICASE, and has been life time member of the Korea Information Society since 1989. His research interests lie in mobile ad hoc networks, real-time computing, and ubiquitous computing.

