Defending Against Collaborative Attacks by Malicious Nodes in MANETs: A Cooperative Bait Detection Approach

Jian-Ming Chang, Po-Chun Tsou, Isaac Woungang, Han-Chieh Chao, and Chin-Feng Lai, *Member, IEEE*

***Abstract*—In mobile ad hoc networks (MANETs), a primary requirement for the establishment of communication among nodes is that nodes should cooperate with each other. In the presence of malevolent nodes, this requirement may lead to serious security concerns; for instance, such nodes may disrupt the routing process. In this context, preventing or detecting malicious nodes launching grayhole or collaborative blackhole attacks is a challenge. This paper attempts to resolve this issue by designing a dynamic source routing (DSR)-based routing mechanism, which is referred to as the cooperative bait detection scheme (CBDS), that integrates the advantages of both proactive and reactive defense architectures. Our CBDS method implements a reverse tracing technique to help in achieving the stated goal. Simulation results are provided, showing that in the presence of malicious-node attacks, the CBDS outperforms the DSR, 2ACK, and best-effort fault-tolerant routing (BFTR) protocols (chosen as benchmarks) in terms of packet delivery ratio and routing overhead (chosen as performance metrics).**

***Index Terms*—Cooperative bait detection scheme (CBDS), collaborative bait detection, collaborative blackhole attacks, detection mechanism, dynamic source routing (DSR), grayhole attacks, malicious node, mobile ad hoc network (MANET).**

I. Introduction

DUE to the widespread availability of mobile devices, mobile ad hoc networks (MANETs) [1], [2] have been widely used for various important applications such as military crisis operations and emergency preparedness and response operations. This is primarily due to their infrastructureless property. In a MANET, each node not only works as a host but can also act as a router. While receiving data, nodes also need cooperation with each other to forward the data packets, thereby forming a wireless local area network [3]. These great features also come with serious drawbacks from a security point of view. Indeed, the aforementioned applications impose some stringent constraints on the security of the network topology, routing, and data traffic. For instance, the presence and collaboration of malicious nodes in the network may disrupt the routing process, leading to a malfunctioning of the network operations.

Many research works have focused on the security of MANETs. Most of them deal with prevention and detection approaches to combat individual misbehaving nodes. In this regard, the effectiveness of these approaches becomes weak when multiple malicious nodes collude together to initiate a collaborative attack, which may result to more devastating damages to the network.

The lack of any infrastructure added with the dynamic topology feature of MANETs make these networks highly vulnerable to routing attacks such as blackhole and grayhole (known as variants of blackhole attacks). In blackhole attacks (see Fig. 1), a node transmits a malicious broadcast informing that it has the shortest path to the destination, with the goal of intercepting

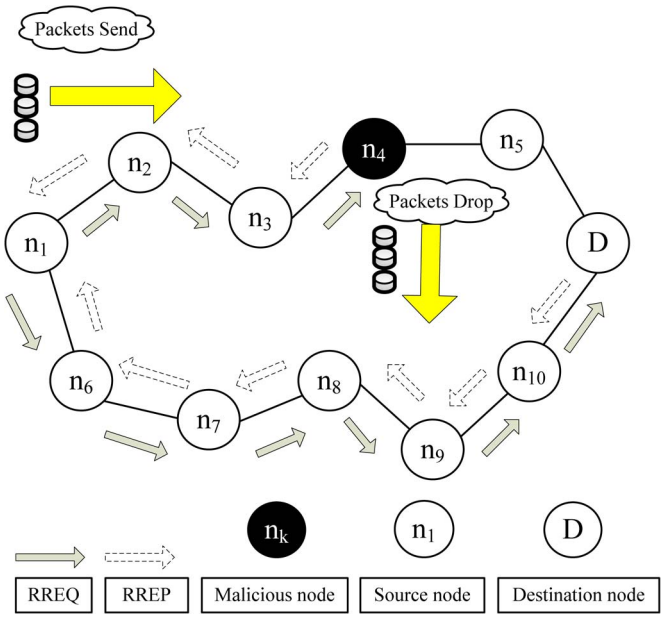

Fig. 1. Blackhole attack–node n_4 drops all the data packets.

Manuscript received May 28, 2012; revised March 19, 2013 and October 16, 2013; accepted December 10, 2013. Date of publication January 9, 2014; date of current version March 2, 2015. An abridged version of this work has been published in the Proceedings of the 2nd International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace and Electronic Systems Technology (VITAE 2011), Chenai, India, February 28–March 3, 2011. This work was supported in part by a grant from the National Science Council of Taiwan, held by the fourth author, under Contract NSC98-2221-E-197-009-MY3 and a grant from the Natural Sciences and Engineering Research Council of Canada (NSERC), held by the third author, under Grant RGPIN/293233-2011.

J.-M. Chang is with the Chung Shan Institute of Science and Technology, Ministry of National Defense, Taoyuan 325, Taiwan (e-mail: a0128866@gmail.com).

P.-C. Tsou is with the Chung Cheng Institute of Technology, National Defense University, Taoyuan 335, Taiwan (e-mail: s952417@gmail.com).

I. Woungang is with the Department of Computer Science, Ryerson University, Toronto, ON M5B 2K3, Canada (e-mail: iwoungan@scs.ryerson.ca).

H.-C. Chao is with the Institute of Computer Science and Information Engineering, National Ilan University, Ilan 260, Taiwan (e-mail: hcc@niu.edu.tw).

C.-F. Lai is with the Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi 621, Taiwan (e-mail: cinfon@ieee.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSYST.2013.2296197

1932-8184 © 2014 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

messages. In this case, a malicious node (so-called blackhole node) can attract all packets by using forged Route Reply (RREP) packet to falsely claim that "fake" shortest route to the destination and then discard these packets without forwarding them to the destination. In grayhole attacks, the malicious node is not initially recognized as such since it turns malicious only at a later time, preventing a trust-based security solution from detecting its presence in the network. It then selectively discards/forwards the data packets when packets go through it. In this paper, our focus is on detecting grayhole/collaborative blackhole attacks using a dynamic source routing (DSR)-based routing technique.

DSR [4] involves two main processes: route discovery and route maintenance. To execute the route discovery phase, the source node broadcasts a Route Request (RREQ) packet through the network. If an intermediate node has routing information to the destination in its route cache, it will reply with a RREP to the source node. When the RREQ is forwarded to a node, the node adds its address information into the route record in the RREQ packet. When destination receives the RREQ, it can know each intermediary node's address among the route. The destination node relies on the collected routing information among the packets in order to send a reply RREP message to the source node along with the whole routing information of the established route. DSR does not have any detection mechanism, but the source node can get all route information concerning the nodes on the route. In our approach, we make use of this feature.

In this paper, a mechanism [so-called cooperative bait detection scheme (CBDS)] is presented that effectively detects the malicious nodes that attempt to launch grayhole/collaborative blackhole attacks. In our scheme, the address of an adjacent node is used as bait destination address to bait malicious nodes to send a reply RREP message, and malicious nodes are detected using a reverse tracing technique. Any detected malicious node is kept in a blackhole list so that all other nodes that participate to the routing of the message are alerted to stop communicating with any node in that list. Unlike previous works, the merit of CBDS lies in the fact that it integrates the proactive and reactive defense architectures to achieve the aforementioned goal.

II. Related Work

Many research works have investigated the problem of malicious node detection in MANETs. Most of these solutions deal with the detection of a single malicious node or require enormous resource in terms of time and cost for detecting cooperative blackhole attacks. In addition, some of these methods require specific environments [5] or assumptions in order to operate. In general, detection mechanisms that have been proposed so far can be grouped into two broad categories. 1) Proactive detection schemes [6]–[12] are schemes that need to constantly detect or monitor nearby nodes. In these schemes, regardless of the existence of malicious nodes, the overhead of detection is constantly created, and the resource used for detection is constantly wasted. However, one of the advantages of these types of schemes is that it can help in preventing or avoiding an attack in its initial stage. 2) Reactive detection schemes [13]–[15] are those that trigger only when the destination node detects a significant drop in the packet delivery ratio.

Among the above schemes are the ones proposed in [9] and [13], which we considered as benchmark schemes for performance comparison purposes. In [9], Liu *et al.* proposed a 2ACK scheme for the detection of routing misbehavior in MANETs. In this scheme, two-hop acknowledgement packets are sent in the opposite direction of the routing path to indicate that the data packets have been successfully received. A parameter acknowledgment ratio, i.e., R_{ack}, is also used to control the ratio of the received data packets for which the acknowledgment is required. This scheme belongs to the class of proactive schemes and, hence, produces additional routing overhead regardless of the existence of malicious nodes. In [13], Xue and Nahrstedt proposed a prevention mechanism called best-effort fault-tolerant routing (BFTR). Their BFTR scheme uses end-to-end acknowledgements to monitor the quality of the routing path (measured in terms of packet delivery ratio and delay) to be chosen by the destination node. If the behavior of the path deviates from a predefined behavior set for determining "good" routes, the source node uses a new route. One of the drawbacks of BFTR is that malicious nodes may still exist in the new chosen route, and this scheme is prone to repeated route discovery processes, which may lead to significant routing overhead. Our proposed detection scheme takes advantage of the characteristics of both the reactive and proactive schemes to design a DSR-based routing scheme able to detect grayhole/collaborative blackhole attacks in MANETs.

III. Proposed Approach

This paper proposes a detection scheme called the cooperative bait detection scheme (CBDS), which aims at detecting and preventing malicious nodes launching grayhole/collaborative blackhole attacks in MANETs. In our approach, the source node stochastically selects an adjacent node with which to cooperate, in the sense that the address of this node is used as bait destination address to bait malicious nodes to send a reply RREP message. Malicious nodes are thereby detected and prevented from participating in the routing operation, using a reverse tracing technique. In this setting, it is assumed that when a significant drop occurs in the packet delivery ratio, an alarm is sent by the destination node back to the source node to trigger the detection mechanism again. Our CBDS scheme merges the advantage of proactive detection in the initial step and the superiority of reactive response at the subsequent steps in order to reduce the resource wastage.

CBDS is DSR-based. As such, it can identify all the addresses of nodes in the selected routing path from a source to destination after the source has received the RREP message. However, the source node may not necessary be able to identify which of the intermediate nodes has the routing information to the destination or which has the reply RREP message or the malicious nodeâ reply forged RREP. This scenario may result in having the source node sending its packets through the fake shortest path chosen by the malicious node, which may then lead to a blackhole attack. To resolve this issue, the function of HELLO message is added to the CBDS to help each node

TABLE I
PACKET FORMAT OF RREQ′

Option Type	Opt Data Len	Request ID
Target Address (RREQ' : Bait address)		
Address[1]		
Address[2]		
.....		
Address[n]		

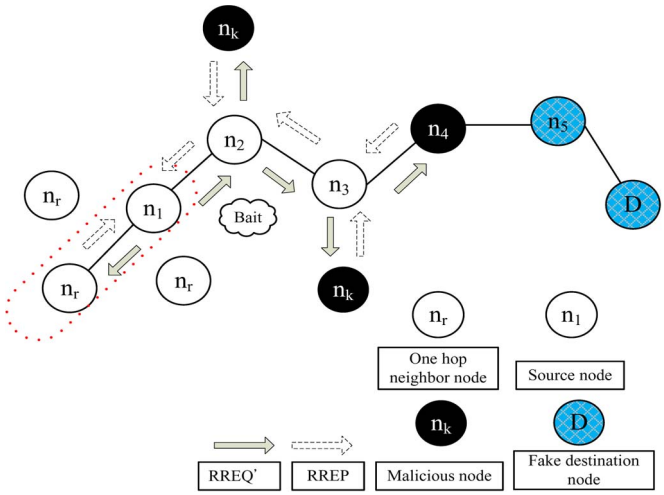

Fig. 2. Random selection of a cooperative bait address.

in identifying which nodes are their adjacent nodes within one hop. This function assists in sending the bait address to entice the malicious nodes and to utilize the reverse tracing program of the CBDS to detect the exact addresses of malicious nodes. The baiting RREQ packets are similar to the original RREQ packets, except that their destination address is the bait address. The modified packet format is shown in Table I.

The CBDS scheme comprises three steps: 1) the initial bait step; 2) the initial reverse tracing step; and 3) the shifted to reactive defense step, i.e., the DSR route discovery start process. The first two steps are initial proactive defense steps, whereas the third step is a reactive defense step.

A. Initial Bait Step

The goal of the bait phase is to entice a malicious node to send a reply RREP by sending the bait RREQ′ that it has used to advertise itself as having the shortest path to the node that detains the packets that were coverted. To achieve this goal, the following method is designed to generate the destination address of the bait RREQ′.

The source node stochastically selects an adjacent node, i.e., n_r, within its one-hop neighborhood nodes and cooperates with this node by taking its address as the destination address of the bait RREQ′. Since each baiting is done stochastically and the adjacent node would be changed if the node moved, the bait would not remain unchanged. This is illustrated in Fig. 2, The bait phase is activated whenever the bait RREQ′ is sent prior to seeking the initial routing path. The follow-up bait phase analysis procedures are as follows.

First, if the n_r node had not launched a blackhole attack, then after the source node had sent out the RREQ′, there would be other nodes' reply RREP in addition to that of the n_r node. This indicates that the malicious node existed in the reply routing, as shown in Fig. 2. Therefore, the reverse tracing program in the next step would be initiated in order to detect this route. If only the n_r node had sent the reply RREP, it means that there was no other malicious node present in the network and that the CBDS had initiated the DSR route discovery phase.

Second, if n_r was the malicious node of the blackhole attack, then after the source node had sent the RREQ′, other nodes (in addition to the n_r node) would have also sent reply RREPs. This would indicate that malicious nodes existed in the reply route. In this case, the reverse tracing program in the next step would be initiated to detect this route. If n_r deliberately gave no reply RREP, it would be directly listed on the blackhole list by the source node. If only the n_r node had sent a reply RREP, it would mean that there was no other malicious node in the network, except the route that n_r had provided; in this case, the route discovery phase of DSR will be started. The route that n_r provides will not be listed in the choices provided to the route discovery phase.

B. Initial Reverse Tracing Step

The reverse tracing program is used to detect the behaviors of malicious nodes through the route reply to the RREQ′ message. If a malicious node has received the RREQ′, it will reply with a false RREP. Accordingly, the reverse tracing operation will be conducted for nodes receiving the RREP, with the goal to deduce the dubious path information and the temporarily trusted zone in the route. It should be emphasized that the CBDS is able to detect more than one malicious node simultaneously when these nodes send reply RREPs. Indeed, when a malicious node, for example, n_m, replies with a false RREP, an address list $P = \{n_1, \ldots n_k, \ldots n_m, \ldots n_r\}$ is recorded in the RREP. If node n_k receives the RREP, it will separate the P list by the destination address n_1 of the RREP in the IP field and get the address list $K_k = \{n_1, \ldots n_k\}$, where K_k represents the route information from source node n_1 to destination node n_k. Then, node n_k will determine the differences between the address list $P = \{n_1, \ldots n_k, \ldots n_m, \ldots n_r\}$ recorded in the RREP and $K_k = \{n_1, \ldots n_k\}$. Consequently, we get

$$K'_k = P - K_k = \{n_{k+1}, \ldots n_m, \ldots n_r\} \tag{1}$$

where K'_k represents the route information to the destination node (recorded after node n_k). The operation result of K'_k is stored in the RREP's "Reserved field" and then reverted to the source node, which would receive the RREP and the address list K'_k of the nodes that received the RREP. To avoid interference by malicious nodes and to ensure that K'_k does not come from malicious nodes, if node n_k received the RREP, it will compare:

1) A. the source address in the IP fields of the RREP;
2) B. the next hop of n_k in the $P = \{n_1, \ldots n_k, \ldots n_m, \ldots n_r\}$;
3) C. one hop of n_k.

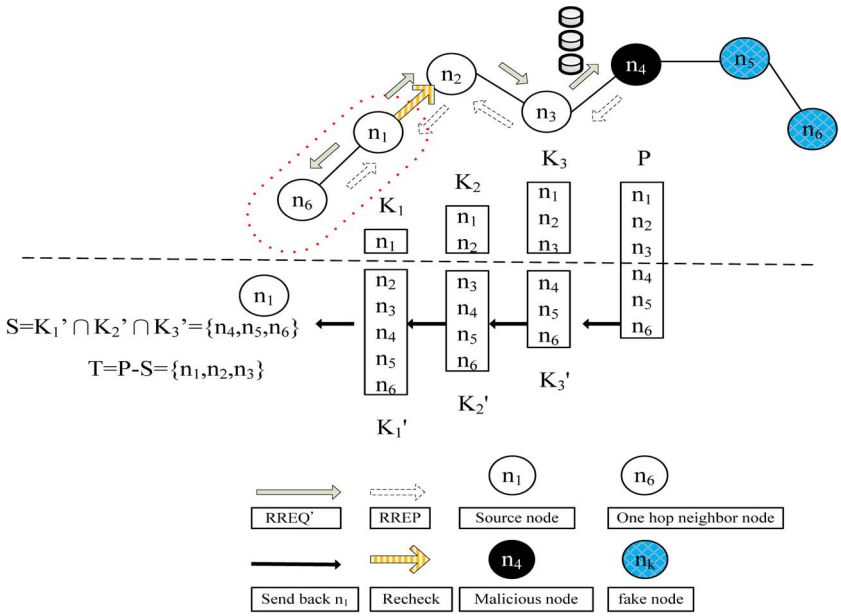

Fig. 3. Reverse tracing program of the CBDS approach.

If A is not the same with B and C, then the received K'_k can perform a forward back. Otherwise, n_k should just forward back the K'_k that was produced by itself.

In Fig. 3, although n_4 can reply with $K'_4 = \{n_5, n_6\}$, n_3 will check and then remove K'_4 when it receives the RREP. After the source node obtains the intersection set of K'_k, the dubious path information S replied by malicious nodes could be detected, i.e.,

$$S = K'_1 \cap K'_2 \cap K'_3 \ldots \cap K'_k. \tag{2}$$

Given that a malicious node would reply the RREP to every RREQ, nodes that are present in a route before this action happened are assumed to be trusted. The set difference operation of P and S is conducted to acquire a temporarily trusted set T, i.e.,

$$T = P - S. \tag{3}$$

To confirm that the malicious node is in set S, the source node would send the test packets to this route and would send the recheck message to the second node toward the last node in T. This requires that the node had entered a promiscuous mode in order to listen to which node the last node in T sent the packets to and fed the result back to the source node. The source node would then store the node in a blackhole list and broadcast the alarm packets through the network to inform all other nodes to terminate their operation with this node. If the last node had dropped the packets instead of diverting them, the source node would store it in the blackhole list. The situations faced by malicious nodes in the route are illustrated in Fig. 3. In this case, a single malicious node n_4 exist in the route, the source node n_1 pretends to send a packet to the destination node n_6. After n_1 sends the RREQ′, node n_4 replies with a false RREP along with the address list $P = \{n_1, n_2, n_3, n_4, n_5, n_6\}$. Here, node n_5 is a random node filled in by n_4. If n_3 had receive the replied RREP by n_4, it would separate the P list by the destination address n_1 of the RREP in the IP field and get the address list $K_3 = \{n_1, n_2, n_3\}$. It would then conduct the set difference operation between the address lists P and $K_3 = \{n_1, n_2, n_3\}$ to acquire $K'_3 = P - K_3 = \{n_4, n_5, n_6\}$, and would reply with the K'_3 and RREP to the source node n_1 according to the routing information in P. Likewise, n_2 and n_1 would perform the same operation after receiving the RREP; will obtain $K'_2 = \{n_3, n_4, n_5, n_6\}$ and $K'_1 = \{n_2, n_3, n_4, n_5, n_6\}$, respectively; and then will send them back to the source node for intersection. The dubious path information of the malicious node, i.e., $S = K'_1 \cap K'_2 \cap K'_3 = \{n_4, n_5, n_6\}$, is obtained. The source node then calculates $P - S = T = \{n_1, n_2, n_3\}$ to acquire a temporarily trusted set. Finally, the source node will send the test packets to this path and the recheck message to n_2, requesting it to enter the promiscuous mode and listening to n_3. As the result of the listening phase, it could be found that n_3 might divert the packets to the malicious node n_4; hence, n_2 would revert the listening result to the source node n_1, which would record n_4 in a blackhole list.

In Fig. 3, if there was a single malicious node n_4 in the route, which responded with a false RREP and the address list $P = \{n_1, n_2, n_3, n_5, n_4, n_6\}$, then this node would have deliberately selected a false node n_5 in the RREP address list to interfere with the follow-up operation of the source node. However, the source node would have to intersect the received K'_k to obtain $S = K'_1 \cap K'_2 \cap K'_3 = \{n_5, n_4, n_6\}$ and $T = P - S = \{n_1, n_2, n_3\}$ and request n_2 to listen to the node that n_3 might send the packets to. As the result of this listening phase, the packets that should have been diverted to n_5 by n_3 should have been sent to n_4. The source node would then store this node to the blackhole list. It is worth mentioning that even if the malicious node cooperated with a false interfering RREP, it would still be detected by the CBDS. In Fig. 3, if n_5 and n_4 were cooperative malicious nodes, we would obtain $T =$

TABLE II
DYNAMIC THRESHOLD ALGORITHM

Dynamic Threshold Algorithm

```
01  double threshold=0.9;
02  InitialProactiveDefense( );
03  double Dynamic(threshold)
04  {   double T1, T2;
05          T1=calculate the time of PDR down to threshold;
06          if(PDR < threshold)
07          InitialProactiveDefense( );
08          T2=calculate the time of PDR down to threshold;
09          if(T2 < T1){
10            if(threshold < 0.95)
11            threshold=threshold+0.01;
12          }
13          else{
14              if(threshold > 0.85)
15              threshold=threshold-0.01;
16          }
17          if(SimulationTime < 800){
18          return threshold;
19          Dynamic(threshold);
20          }
21          else
22          return 0.9;
23   }
24
```

$P - S = \{n_1, n_2, n_3\}$, and n_2 would be requested to listen to which node n_3 might send the packets. Either n_5 or n_4 would be detected, and their cooperation stopped. Hence, the remaining nodes would be baited and detected. Fig. 2 illustrates that even if there were more malicious nodes in MANETs, the CBDS would still detect them simultaneously when they send the reply RREP.

C. Shifted to Reactive Defense Phase

After the above initial proactive defense (steps A and B), the DSR route discovery process is activated. When the route is established and if at the destination it is found that the packet delivery ratio significantly falls to the threshold, the detection scheme would be triggered again to detect for continuous maintenance and real-time reaction efficiency. The threshold is a varying value in the range [85%, 95%] that can be adjusted according to the current network efficiency. The initial threshold value is set to 90%.

We have designed a dynamic threshold algorithm (see Table II) that controls the time when the packet delivery ratio falls under the same threshold. If the descending time is shortened, it means that the malicious nodes are still present in the network. In that case, the threshold should be adjusted upward. Otherwise, the threshold will be lowered.

The operations of the CBDS are captured in Fig. 4. It should be noticed that the CBDS offers the possibility to obtain the dubious path information of malicious nodes as well as that of trusted nodes; thereby, it can identify the trusted zone by simply looking at the malicious nodes reply to every RREP. In addition, the CBDS is capable of observing whether a malicious node would drop the packets or not. As a result, the proportion of dropped packets is disregarded, and malicious nodes launching a grayhole attack would be detected by the CBDS the same way as those launching blackhole attacks are detected.

IV. PERFORMANCE EVALUATION

A. Simulation Parameters

The QualNet 4.5 simulation tool [16] is used to study the performance of our CBDS scheme. We employ the IEEE 802.11 [17] MAC with a channel data rate of 11 Mb/s. In our simulation, the CBDS default threshold is set to 90%. All remaining simulation parameters are captured in Table III. The network used for our simulations is depicted in Fig. 5; and we randomly select the malicious nodes to perform attacks in the network.

B. Performance Metrics

We have compared the CBDS against the DSR [4], 2ACK [9], and BFTR [13] schemes, chosen as benchmarks, on the basis of the following performance metrics.

1) **Packet Delivery Ratio:** This is defined as the ratio of the number of packets received at the destination and the number of packets sent by the source. Here, $pktd_i$ is the number of packets received by the destination node in the ith application, and $pkts_i$ is the number of packets sent by the source node in the ith application. The average packet delivery ratio of the application traffic n, which is denoted by PDR, is obtained as

$$PDR = \frac{1}{n}\sum_{i=1}^{n}\frac{pktd_i}{pkts_i}. \tag{4}$$

2) **Routing Overhead:** This metric represents the ratio of the amount of routing-related control packet transmissions to the amount of data transmissions. Here, cpk_i is the number of control packets transmitted in the ith application traffic, and pkt_i is the number of data packets transmitted in the ith application traffic. The average routing overhead of the application traffic n, which is denoted by RO, is obtained as

$$RO = \frac{1}{n}\sum_{i=1}^{n}\frac{cpk_i}{pkt_i}. \tag{5}$$

3) **Average End-to-End Delay:** This is defined as the average time taken for a packet to be transmitted from the source to the destination. The total delay of packets received by the destination node is d_i, and the number of packets received by the destination node is $pktd_i$. The average end-to-end delay of the application traffic n, which is denoted by E, is obtained as

$$E = \frac{1}{n}\sum_{i=1}^{n}\frac{d_i}{pktd_i}. \tag{6}$$

4) **Throughput:** This is defined as the total amount of data (b_i) that the destination receives them from the source divided by the time (t_i) it takes for the destination to get the final packet. The throughput is the number of bits transmitted per second. The throughput of the application traffic n, which is denoted by T, is obtained as

$$T = \frac{1}{n}\sum_{i=1}^{n}\frac{b_i}{t_i}. \tag{7}$$

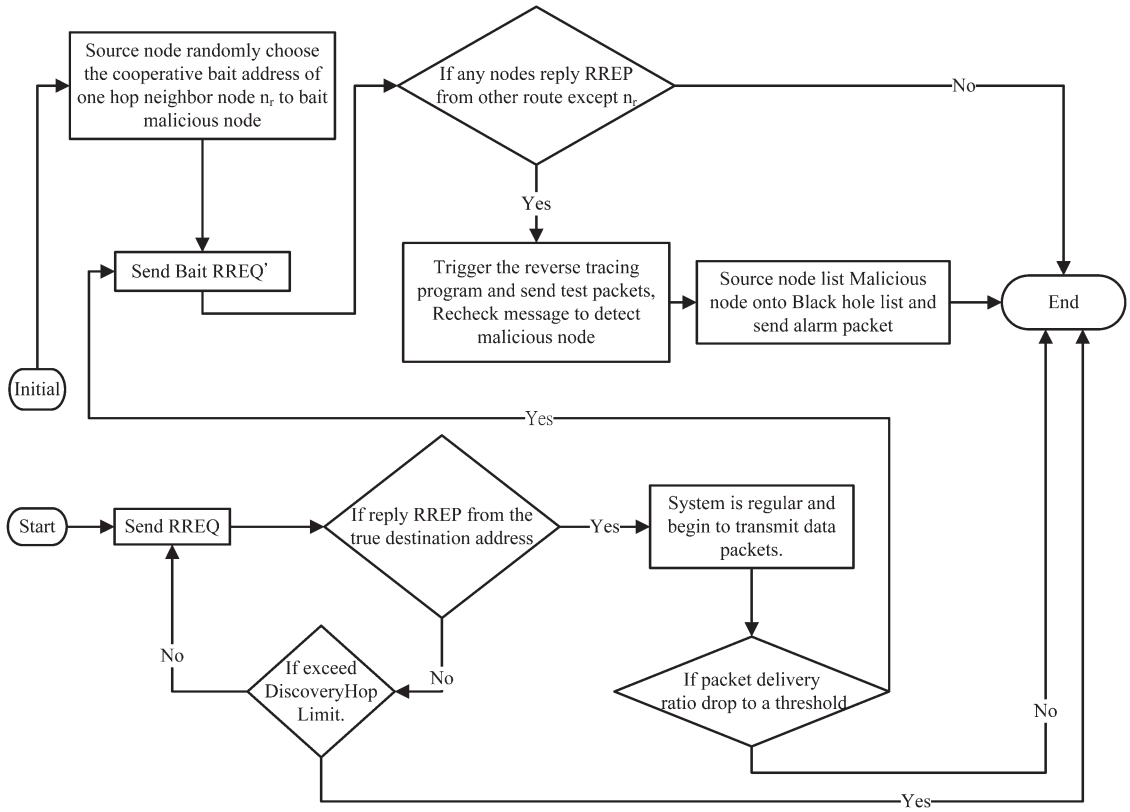

Fig. 4. Operations of the CBDS.

TABLE III
SIMULATION PARAMETERS

Parameter	Value
Application traffic	*10 CBR*
Transmission rate	*4 packets/s*
Radio range	*250m*
Packet size	*512 bytes*
Channel data rate	*11Mbps*
Pause time	*0s*
Maximum Speed	*20m/s*
Simulation time	*800s*
Number of nodes	*50*
Area	*700m*700m*
Malicious nodes	*0% 40%*
Threshold	*Dynamic threshold*

Two simulation scenarios are considered:

1) Scenario 1: Varying the percentage of malicious nodes with a fixed mobility.
2) Scenario 2: Varying the mobility of nodes under fixed percentage of malicious nodes.

Under these scenarios, we study the effect of different thresholds of the CBDS on the aforementioned performance parameters. The results are as follows.

C. Varying the Percentage of Malicious Nodes With a Fixed Mobility

First, we study the packet delivery ratio of the CBDS and DSR for different thresholds when the percentage of malicious nodes in the network varies from 0% to 40%. The maximum speed of nodes is set to 20 m/s. Here, the threshold value is set to 85%, 95%, and the dynamic threshold, respectively. The results are captured in Fig. 6. In Fig. 6, it can be observed that DSR drastically suffers from blackhole attacks when the percentage of malicious nodes increases. This is attributed to the fact that DSR has no secure method for detecting/preventing blackhole attacks. Our CBDS scheme shows a higher packet delivery ratio compared with that of DSR. Even in the case where 40% of the total nodes in the network are malicious, the CBDS scheme still successfully detects those malicious nodes while keeping the packet delivery ratio above 90%. A threshold of 95% would then result in earlier route detection than when the threshold is 85% or is set to the dynamic threshold value. Thus, the packet delivery ratio when using a threshold of 95% is higher than that obtained when using a threshold of 85% or the dynamic threshold.

Second, we study the routing overhead of the CBDS and DSR for different thresholds. The results are captured in Fig. 7. In Fig. 7, it can be observed that when the number of malicious nodes increases, DSR produces the lowest routing overhead compared with the CBDS. This is attributed to the fact that DSR has no intrinsic security method or defensive mechanism. In fact, the routing overhead produced by the CBDS for different thresholds is a little bit higher than that produced by DSR; this might be due to the fact that the CBDS would first send bait packets in its initial bait phase and then turn into a reactive defensive phase afterward. Consequently, a tradeoff should be made between routing overhead and packet delivery ratio. We have studied the effect of thresholds on the routing overhead. As expected, it was found that the routing overhead of the CBDS reaches the highest value when the threshold is set to

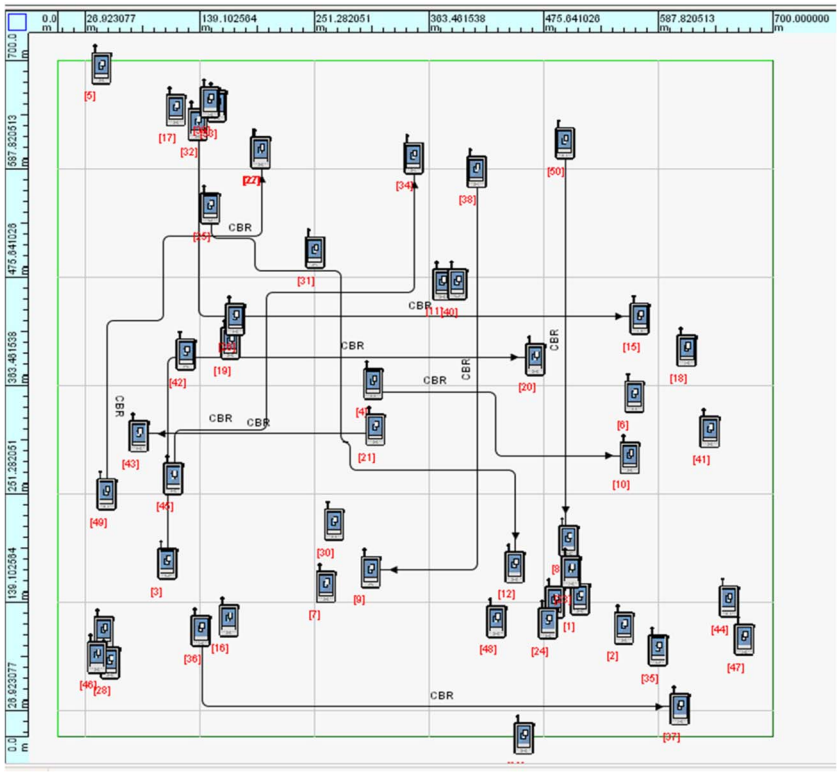

Fig. 5. Network topology.

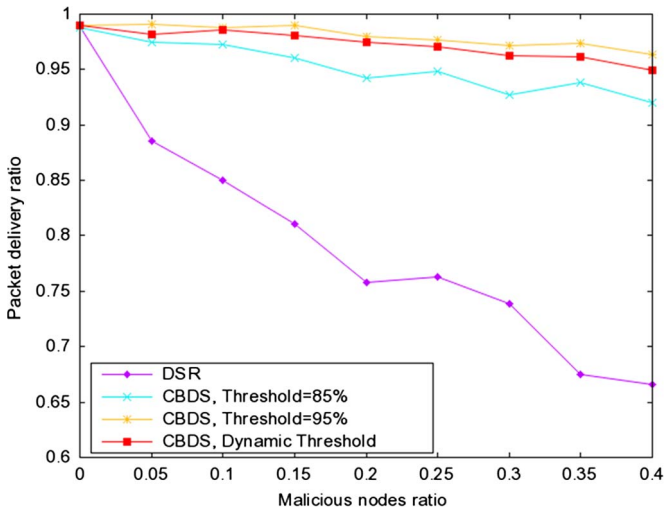

Fig. 6. Packet delivery ratio of DSR and the CBDS for different thresholds.

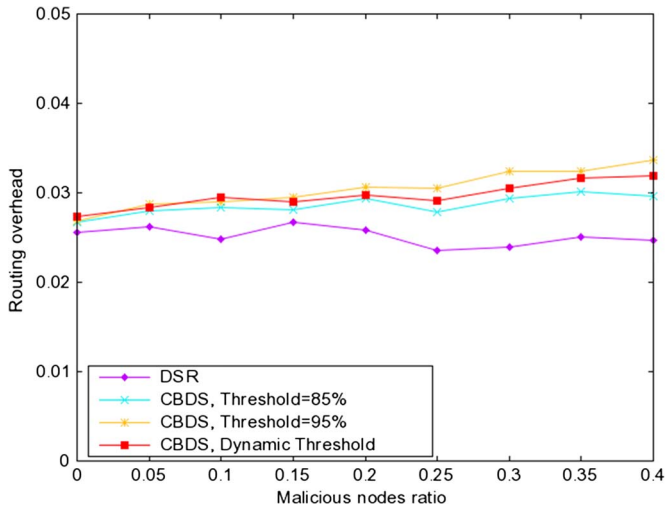

Fig. 7. Routing overhead of DSR and the CBDS for different thresholds.

95%. This is attributed to the fact that the detection scheme of CBDS triggers fast when the threshold value is 95% compared with when it is set to 85% or when it is equal to the dynamic threshold value. Thus, the bait packets will be sent many times in the network. It should be noticed that the dynamic threshold value can be adjusted according to the network performance.

Third, we study the end-to-end delay of the CBDS and DSR for different thresholds. The results are captured in Fig. 8. In Fig. 8, it can be observed that the CBDS incurs a little bit more end-to-end delay compared with that of DSR. This is attributed to the fact that the CBDS necessitated more time to bait and detect malicious nodes. Therefore, a tradeoff must be made between end-to-end delay and packet delivery ratio. Even in the case that there are more malicious nodes in the network, the CBDS would still detect them simultaneously when they reply with a RREP. Thus, the end-to-end delay of the CBDS for different thresholds does not increase when the number of malicious nodes increases. We further study the effect of thresholds on the end-to-end delay. Although a threshold of 85% produces the shortest delay, the resulting packet delivery ratio appears to be lower than that produced when the threshold is set to 95% or is set to the dynamic threshold value.

Fourth, we study the throughput of the CBDS and DSR for different thresholds. The results are captured in Fig. 9. In Fig. 9, it can be observed that DSR suffers the most from malicious-node attacks compared with the CBDS. In addition, the CBDS

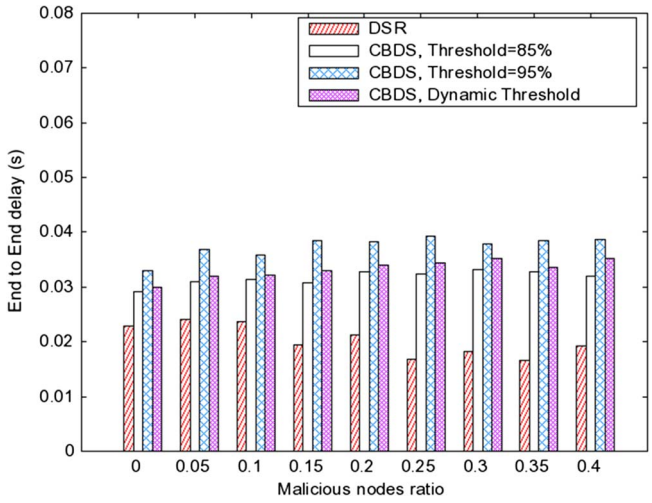

Fig. 8. End-to-end delay of DSR and the CBDS for different thresholds.

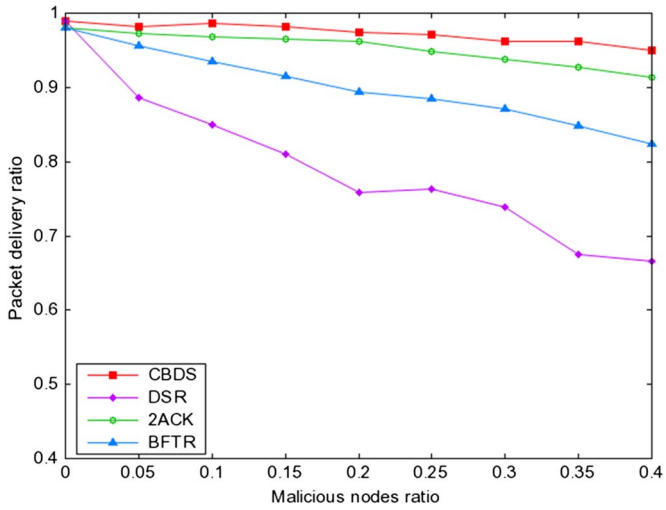

Fig. 10. Effect of malicious nodes on the packet delivery ratio.

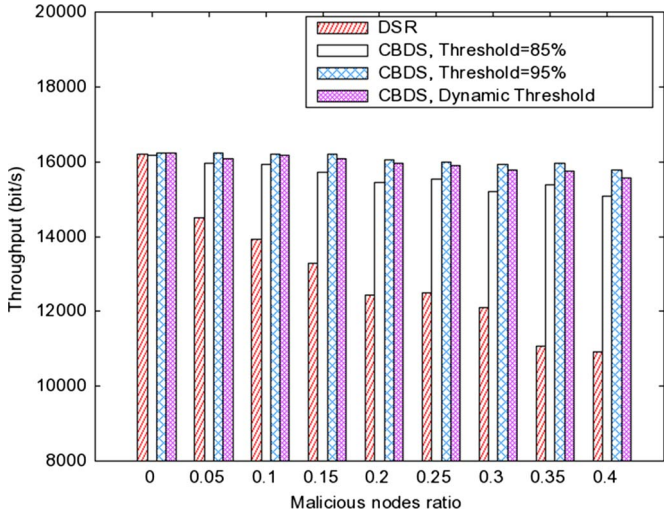

Fig. 9. Throughput of DSR and the CBDS for different thresholds.

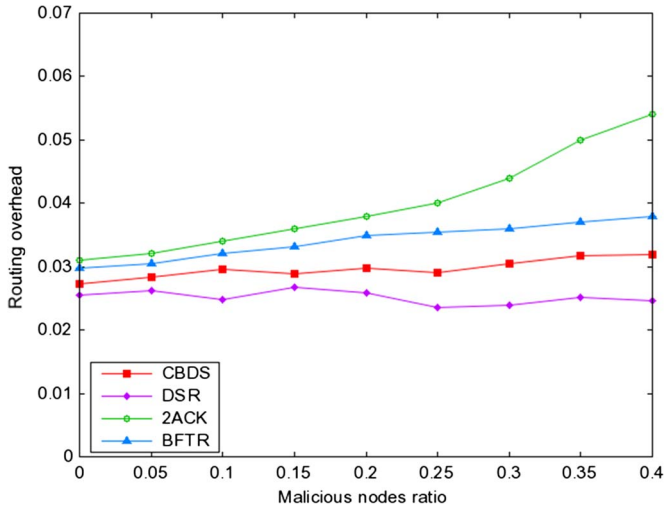

Fig. 11. Effect of malicious nodes on the routing overhead.

with different thresholds results in higher throughput than DSR. We further study the effect of thresholds on the throughput. The results are shown in Fig. 10. In Fig. 10, it can be observed that the throughput obtained when the threshold is set to 95% is, in general, slightly higher than that obtained when the threshold is set to 85% or is set to the dynamic threshold value. Even in the case where the number of malicious nodes present in the network is relatively high (up to 40%), it is observed that the CBDS can still detect malicious nodes successfully while keeping the throughput above 15 000 bit/s.

Fifth, we compare DSR, 2ACK, BFTR, and CBDS in terms of packet delivery ratio and routing overhead when the malicious nodes increase in the network. Here, the threshold for the CBDS is set to the dynamic threshold value. The results are captured in Figs. 10 and 11, respectively.

In Fig. 10, it can also be observed that DSR heavily suffers from increasing blackhole attacks since it does not have any detection and protection mechanism to prevent blackhole attacks. When the percentage of malicious nodes varies in the network from 0% to 40%, BFTR does not detect malicious nodes directly. It chooses a new route that may still include malicious nodes when the end-to-end performance of a route deviates from the predefined behavior of good routes. Therefore, the packet delivery ratio of BFTR is lower than that observed for both the 2ACK and CBDS schemes. Moreover, the packet delivery ratio of the CBDS is highest compared with that of DSR. This is attributed to the fact that the CBDS sends bait packets to bait malicious nodes when replying and is capable of tracing the location of the blackhole node at the initial stage.

In Fig. 11, it can be observed that when the percentage of malicious nodes increases, DSR produces the lowest routing overhead compared with all other schemes including the CBDS. This is attributed to the fact that DSR has no intrinsic security or defensive mechanism. Moreover, the CBDS is able to achieve proactive detection in the initial stage and then change into reactive response in the later stage. Through this feature, the advantage of proactive detection and the superiority of reactive response can be merged to reduce the waste of resource. This has led to a better routing overhead for the CBDS compared with that of the 2ACK and BFTR schemes. Furthermore, the 2ACK scheme has the highest routing overhead compared with that of BTFR and CBDS. This is attributed to the fact that

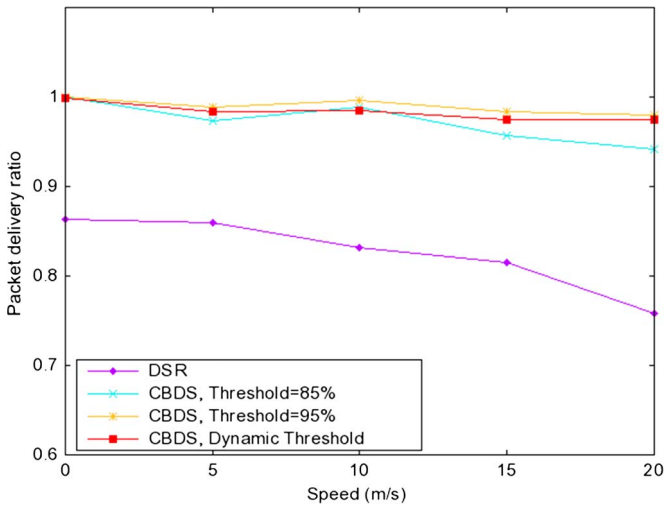

Fig. 12. Packet delivery ratio for different thresholds, under varying node speed.

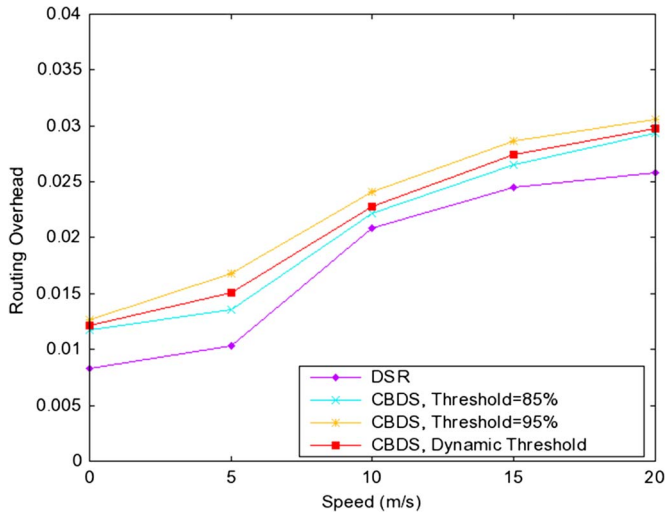

Fig. 13. Routing overhead for different thresholds, under varying node speed.

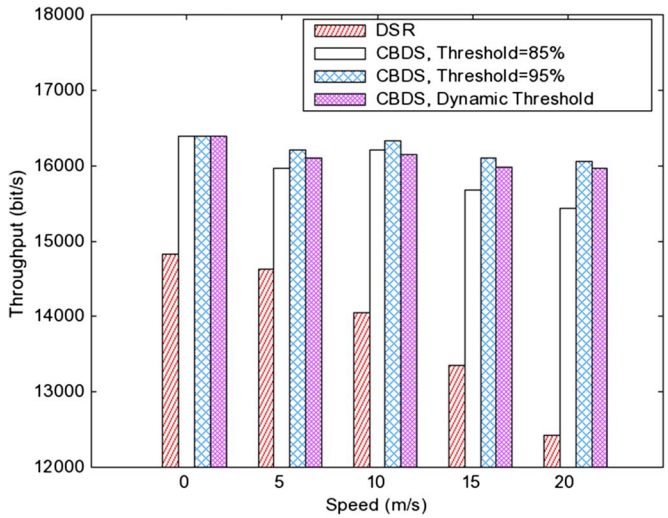

Fig. 14. Throughput for different thresholds, under varying node speed.

2ACK is a proactive scheme, which incurs routing overhead regardless of the existence of malicious nodes. Although BFTR belongs to the family of reactive schemes, the new route that it has selected may still have malicious nodes in it, which, in turn, will trigger repeated route discovery processes, causing the additional routing overhead observed in BFTR compared with the CBDS.

D. Varying the Mobility of Nodes Under a Fixed Percentage of Malicious Nodes

In this scenario, the maximum speed of nodes is varied from 0 to 20 m/s, and the percentage of malicious nodes is fixed to 20%.

First, we study the packet delivery ratio of the CBDS and DSR for different thresholds. The threshold value is set to 85%, 95%, and the dynamic threshold, respectively. The results are captured in Fig. 12. It can also be observed that the packet delivery ratio of DSR and the CBDS for different thresholds slightly decreases when the node's speed increases. The CBDS yields a higher packet delivery ratio compared with DSR. Finally, the CBDS can detect malicious nodes successfully while keeping the packet delivery ratio above 90%.

Second, we study the routing overhead of the CBDS and DSR for different thresholds. The threshold value is set to 85%, 95%, and the dynamic threshold, respectively. The results are captured in Fig. 13. In Fig. 13, it can be observed that the routing overhead of DSR and the CBDS for different thresholds increases when the node's speed increases. Moreover, the CBDS can still detect malicious nodes successfully while keeping a routing overhead a little higher than that of DSR.

Third, we study the throughput of the CBDS and DSR for different thresholds. The threshold value is set to 85%, 95%, and the dynamic threshold, respectively. The results are captured in Fig. 14. In Fig. 14, it can be observed that the throughput of DSR and the CBDS for different thresholds slightly decreases when the node's speed increases. The CBDS yields the highest throughput compared with DSR in all cases. It is also found that the CBDS can still keep the highest throughput while avoiding interference with malicious nodes.

Fourth, we study the end-to-end delay of the CBDS and DSR for different thresholds. The threshold value is set to 85%, 95%, and the dynamic threshold, respectively. The results are captured in Fig. 15. In Fig. 15, it can be observed that the average end-to-end delay incurred by the CBDS is higher than that incurred by DSR in all cases. This is attributed to the fact that the CBDS requires more time to detect and trace the malicious nodes, which is not the case for DSR since the latter has no intrinsic malicious node detection mechanism.

V. Conclusion

In this paper, we have proposed a new mechanism (called the CBDS) for detecting malicious nodes in MANETs under gray/collaborative blackhole attacks. Our simulation results revealed that the CBDS outperforms the DSR, 2ACK, and BFTR schemes, chosen as benchmark schemes, in terms of routing overhead and packet delivery ratio. As future work, we intend to 1) investigate the feasibility of adjusting our CBDS approach

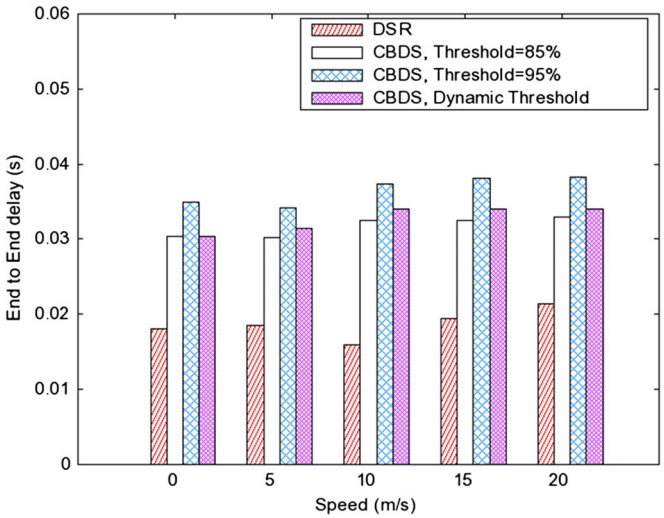

Fig. 15. End-to-end delay for different thresholds, under varying node speed.

to address other types of collaborative attacks on MANETs and to 2) investigate the integration of the CBDS with other well-known message security schemes in order to construct a comprehensive secure routing framework to protect MANETs against miscreants.

REFERENCES

[1] P.-C. Tsou, J.-M. Chang, H.-C. Chao, and J.-L. Chen, "CBDS: A cooperative bait detection scheme to prevent malicious node for MANET based on hybrid defense architecture," in *Proc. 2nd Intl. Conf. Wireless Commun., VITAE*, Chenai, India, Feb. 28–Mar., 03, 2011, pp. 1–5.

[2] S. Corson and J. Macker, RFC 2501, Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, Jan. 1999. (Last retrieved March 18, 2013). [Online]. Available: http://www.elook.org/computing/rfc/rfc2501.html

[3] C. Chang, Y. Wang, and H. Chao, "An efficient Mesh-based core multicast routing protocol on MANETs," *J. Internet Technol.*, vol. 8, no. 2, pp. 229–239, Apr. 2007.

[4] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile Comput.*, pp. 153–181, 1996.

[5] I. Rubin, A. Behzad, R. Zhang, H. Luo, and E. Caballero, "TBONE: A mobile-backbone protocol for ad hoc wireless networks," in *Proc. IEEE Aerosp. Conf.*, 2002, vol. 6, pp. 2727–2740.

[6] A. Baadache and A. Belmehdi, "Avoiding blackhole and cooperative blackhole attacks in wireless ad hoc networks," *Intl. J. Comput. Sci. Inf. Security*, vol. 7, no. 1, 2010.

[7] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proc. 6th Annu. Intl. Conf. MobiCom*, 2000, pp. 255–265.

[8] K. Vishnu and A. J Paul, "Detection and removal of cooperative black/gray hole attack in mobile ad hoc networks," *Int. J. Comput. Appl.*, vol. 1, no. 22, pp. 28–32, 2010.

[9] K. Liu, D. Pramod, K. Varshney, and K. Balakrishnan, "An Acknowledgement based approach for the detection of routing misbehavior in MANETs," *IEEE Trans. Mobile Comput.*, vol. 6, no. 5, pp. 536–550, May 2007.

[10] H. Deng, W. Li, and D. Agrawal, "Routing security in wireless ad hoc network," *IEEE Commun. Mag.*, vol. 40, no. 10, Oct. 2002.

[11] S. Ramaswamy, H. Fu, M. Sreekantaradhya, J. Dixon, and K. Nygard, "Prevention of cooperative blackhole attacks in wireless ad hoc networks," in *Proc. Int. Conf. Wireless Netw.*, Jun. 2003, pp. 570–575.

[12] H. Weerasinghe and H. Fu, "Preventing cooperative blackhole attacks in mobile ad hoc networks: Simulation implementation and evaluation," in *Proc. IEEE ICC*, 2007, pp. 362–367.

[13] Y. Xue and K. Nahrstedt, "Providing fault-tolerant ad hoc routing service in adversarial environments," *Wireless Pers.Commun.*, vol. 29, pp. 367–388, 2004.

[14] W. Kozma and L. Lazos, "REAct: resource-efficient accountability for node misbehavior in ad hoc networks based on random audits," in *Proc. WiSec*, 2009, pp. 103–110.

[15] W. Wang, B. Bhargava, and M. Linderman, "Defending against collaborative packet drop attacks on MANETs," in *Proc.* 28^th *IEEE Int. Symp. Reliable Distrib. Syst.*, New Delhi, India, Sep. 2009.

[16] QualNet Simulaton Tool, Scalable Network Technologies. (Last retrieved March 18, 2013). [Online]. Available: http://www.qualnet.com

[17] *IEEE Standard for Information Technology*, IEEE Std 802.11-14997, 1997, Telecommunications and Information exchange between systems: wireless LAN medium access control (MAC) and physical layer (PHY) Specifications, pp. i-445.

Jian-Ming Chang received the M.S. degree in electrical engineering and the Ph.D. degree in computer science and information engineering from National Dong Hwa University, Hualien, Taiwan, in 2007 and 2012, respectively.

He is currently an Assistant Researcher with the Electronic System Research Division, Chung-Shan Institute of Science and Technology, Ministry of National Defense, Taoyuan, Taiwan. His research interests focus on the next-generation Internet, mobile computing, cellular mobility management, personal communication networks, adaptive antenna arrays, beamforming, and phased-array radar systems.

Po-Chun Tsou received the B.S. degree in computer science and engineering from Chung Cheng Institute of Technology, National Defense University, Taoyuan, Taiwan, in 2006 and the M.S. degrees in computer science and information engineering from National Ilan University, Ilan, Taiwan, in 2011.

He is currently an R&D officer with the Chung Cheng Institute of Technology, National Defense University. His research interests include wireless networks, mobile computing, and information security.

Isaac Woungang received the M.Sc. degree in mathematics from the Université de la Méditerranée-Aix Marseille II, Luminy, France, in 1990; the Ph.D. degree in mathematics from the Université du Sud, Toulon-Var, France, in 1994; and the M.A.Sc. degree from the INRS-Énergie, Matériaux et Télécommunications, University of Quebec, Montreal, QC, Canada, in 1999.

From 1999 to 2002, he was a Software Engineer with Nortel Networks. Since 2002, he has been with the Department of Computer Science, Ryerson University, Toronto, ON, Canada. In 2004, he founded the Distributed Applications and Broadband NEtworks Laboratory (DABNEL) R&D group. His research interests include network security, computer communication networks, and mobile communication systems.

Han-Chieh Chao received the M.S. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1989 and 1993, respectively.

He is a jointly appointed Professor with the Department of Electronic Engineering and the Institute of Computer Science and Information Engineering, National Ilan University, Ilan, Taiwan. His research interests include high-speed networks, wireless networks, and IPv6-based networks and applications.

Dr. Chao is also serving as an IPv6 Steering Committee Member and the Deputy Director of the R&D Division of the NICI Taiwan and a Cochair of the Technical Area for IPv6 Forum Taiwan. He is a Fellow of the Institute of Engineering and Technology and the British Computer Society.

Chin-Feng Lai (M'07) received the Ph.D. degree from National Cheng Kung University, Tainan, Taiwan, in 2008.

Since 2013, he has been an Assistant Professor with the Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan. He has more than 100 paper publications. His research focuses on Internet of Things, body sensor networks, E-healthcare, mobile cloud computing, cloud-assisted multimedia networks, and embedded systems.

Dr. Lai is an Associate Editor-in-Chief for the *Journal of Internet Technology* and serves as the Editor or Associate Editor for *IET Networks*. He received the Best Paper Award from the IEEE 10th International Conference on Embedded and Ubiquitous Computing (EUC 2012).