Wireless Netw
DOI 10.1007/s11276-017-1621-z

Active detection in mitigating routing misbehavior for MANETs

Cong Pu[1] · **Sunho Lim**[2] · **Jinseok Chae**[3] · **Byungkwan Jung**[2]

Abstract Mobile ad hoc network (MANET) is vulnerable to security attacks because of the shared radio medium and lack of centralized coordination. Since most multi-hop routing protocols implicitly assume cooperative routing and are not originally designed for security attacks, MANET has been challenged by diverse denial-of-service attacks that often interfere with the protocol and interrupt on-going communication. In this paper, we propose an explore-based active detection scheme, called EBAD, to efficiently mitigate the routing misbehaviors in MANETs running with dynamic source routing. The basic idea is that a source node broadcasts a route request packet with a fictitious destination node to lure potential malicious nodes to reply a fake route reply packet. If the source node receives the fake route reply packet or an intermediate node cannot decrypt the received route reply packet, the routing misbehavior can be detected. We also propose a route expiry timer based approach to reduce the effect of route cache pollution because of the fake route reply. We present a simple analytical model of the EBAD and its numerical result in terms of detection rate. We also conduct extensive simulation experiments using the OMNeT++ for performance evaluation and comparison with the existing schemes, CBDS and 2ACK. The simulation results show that the proposed countermeasure can not only improve the detection rate and packet delivery ratio but also can reduce the energy consumption and detection latency.

Keywords Denial-of-service (DoS) · Dynamic source routing (DSR) · Mobile ad hoc network (MANET) · Routing misbehavior

✉ Cong Pu
puc@marshall.edu

Sunho Lim
sunho.lim@ttu.edu

Jinseok Chae
jschae@inu.ac.kr

Byungkwan Jung
byung.jung@ttu.edu

[1] Division of Computer Science, Marshall University, Huntington, WV 25755, USA

[2] Department of Computer Science, Texas Tech University, Lubbock, TX 79409, USA

[3] Department of Computer Science and Engineering, Incheon National University, Incheon 22012, Korea

1 Introduction

With the rapid advancement of pervasive high speed wireless networks and mobility support, mobile ad hoc network (MANET) has been increasingly popular in deploying diverse military and civilian applications [1]. MANET consists of a set of wireless and mobile nodes (later, nodes) that cooperatively communicate among themselves directly or indirectly via multi-hop relays without the help of a wired infrastructure. A significant volume of research on MANET has been conducted in the past decades, and primarily been focused on developing routing protocols to increase the connectivity among nodes in the presence of constantly varying network topologies. However, due to the shared radio medium and lack of centralized coordination, MANET is exposed to serious security threats. Since nodes are implicitly assumed to operate cooperative routing, a malicious node can easily overhear an on-flying packet and duplicate, corrupt, alter, or even drop any incoming packet. In particular, MANET has been challenged by a denial-of-service (DoS) attack

Published online: 23 November 2017

that primarily targets service availability to diminish the network capacity by disrupting routing protocols or interfering with ongoing communications, rather than subverting the service itself. For example, a malicious node may actively show a routing misbehavior by falsely claiming that it knows the shortest route to a requested destination node, or selectively or randomly dropping any incoming packet on purpose to deafen an intended destination node.

In this paper, we investigate routing misbehaviors causing potential DoS attacks in MANETs running with dynamic source routing (DSR) [2], where malicious nodes falsely claim a fake shortest route to a destination node to attract network traffic on purpose. Unlike prior countermeasures [3–12], where each node passively observes and detects the routing misbehaviors of its neighbor nodes, we propose an active countermeasure and its corresponding techniques to energy-efficiently detect the routing misbehaviors and proactively prevent potential malicious nodes from being involved in the routing path. Our contribution has two parts: (1) We first propose an explore-based active detection scheme, called EBAD, in MANETs running with DSR. The basic idea is that a source node broadcasts a route request packet with a fictitious destination node to lure potential malicious nodes to reply a fake route reply packet. The EBAD is also incorporated with a digital signature technique to detect faulty information in the route reply packet. A route expiry timer is deployed to reduce the effect of route cache pollution caused by the fake route reply. (2) Second, we present a simple analytical model of the EBAD and show its numerical result in terms of detection rate. We also revisit two existing schemes, CBDS [13] and 2ACK [6], and implement them for performance comparison. Here, the original DSR without detection mechanism is used as the lower bound of performance. We develop a customized discrete event-driven simulation framework by using the OMNeT++ [14], conduct extensive simulation experiments, and evaluate the performance in terms of detection rate, energy consumption, packet delivery ratio (PDR), node behavior distribution, exploring probability, and statistics of Data packets. The simulation results indicate that the proposed scheme is a viable approach in mitigating the routing misbehaviors in MANETs running with DSR.

The remainder of this paper is organized as follows. Prior approaches are summarized and analyzed in Sect. 2. The basic DSR operations and their potential vulnerabilities are investigated with a preliminary result in Sect. 3. The proposed countermeasure and its simple analysis are presented in Sect. 4. Section 5 is devoted to extensive simulation experiments and performance comparison and analysis. We further explore the potential extensions of our proposed countermeasure in Sect. 6. Finally, we conclude the paper in Sect. 7.

2 Related work

In this section, we categorize prior schemes in terms of monitor-, acknowledgment-, cryptography-, inducement-based, and other approaches in multi-hop networks and analyze their operations.

Monitor-based approach: The network traffic and communication activities are observed and recorded to detect potential routing misbehaviors. In [7], each node observes both downstream and upstream network traffic of its adjacent nodes and estimates a packet loss rate to detect a selective forwarding attack in wireless mesh networks (WMNs). The [8] is a variant of [7] with addition of a two-hop acknowledgment in the link-layer to detect a collaborative selective forwarding attack. The [10, 11] consider a reputation table to evaluate the routing behaviors of adjacent nodes in wireless sensor networks (WSNs). The [9] proposes a countermeasure to on–off attacks which are specifically designed to disrupt the trust management and redemption schemes. By behaving well and badly alternatively, the on–off attack aims to make the trust management scheme consider a bad behavior as a temporary error. In [15], each node records a set of limited traces of routing operations and exchanges it with its adjacent nodes to identify any routing misbehavior in energy harvesting motivated networks (EHNets).

Acknowledgment-based approach: The key operation is that a set of intermediate nodes located along the forwarding path to a destination node is responsible for sending an explicit message back to a source node to either confirm that a packet has been received or report any routing misbehavior. In [4] and its extension [5], a source node randomly selects multiple checkpoint nodes per packet basis and each checkpoint node replies an acknowledgment (ACK) packet back to the source node in WSNs. In [6], each intermediate node generates and forwards a two-hop ACK packet in the opposite direction of the Data packet to detect a routing misbehavior in MANETs. In the SCAD [12], a light-weight countermeasure to selective forwarding attack is proposed by deploying a single checkpoint node integrated with the timeout and hop-by-hop retransmission techniques.

Cryptography-based approach: A basic encryption method is deployed to implement secure communication. The [16] is designed based on DSR and primarily uses the Timed Efficient Stream Loss-tolerant Authentication (TESLA) [17] as a broadcast authentication protocol that requires a loose time synchronization to secure the route discovery and maintenance procedures. Here, the TESLA is an efficient broadcast authentication protocol with low communication and computation overhead, which can scale to large number of receivers and tolerate potential

packet loss. The [18–20] propose that both source and destination nodes share public or symmetric key information against the adversary that monitors the network traffic.

Inducement-based approach: The basic idea is that a piece of information is hidden or fake information is utilized to lure potential malicious nodes into revealing their routing misbehaviors. In [13], a cooperative bait detection scheme (CBDS) running with DSR is proposed by luring a malicious node to response (i.e., route reply packet) for a false route request (i.e., route request packet) in MANETs. In [21], each node actively pretends not to monitor its adjacent nodes on purpose, but in fact it stealthily observes any routing operation of its adjacent nodes to detect a lurking deep malicious node in EHNets.

Other approach: The [22] conducts a fine-grained analysis (FGA) to investigate the cause of packet loss. The FGA profiles wireless links between nodes and their adjacent nodes by leveraging resident parameters based on the received signal strength and link quality indicators. According to the profiles, the FGA can determine whether a packet loss is caused by an attacker or not. In the AMD [23], an audit-based misbehavior detection approach is proposed to isolate continuous or selective packet droppers. The AMD integrates reputation management, trustworthy route discovery, and identification of misbehaving nodes based on behavioral audits. Node behavior is evaluated based on per-packet basis, without employing energy-expensive overhearing or intensive acknowledgment techniques. The [24] proposes a collaborative detection approach to selective forwarding attack, where each node is monitored and evaluated by its adjacent nodes in two time windows. At the end of each time window, all monitored neighbor nodes are evaluated by an intrusion detection system (IDS) node and if an attack was detected, a voting process is executed to identify a malicious node.

In summary, most prior schemes passively detect the routing misbehaviors witnessed in a vulnerable case by observing and recording adjacent nodes' routing behavior. To the best of our knowledge, little work [13] has been done for an active countermeasure by detecting potential malicious nodes and preventing them from being involved in the routing path in advance. However, due to certain unsolved problems in [13], multiple malicious nodes can collude together and fail the detection process without being detected. In this paper, we propose a novel detection scheme and its corresponding techniques to efficiently detect the routing misbehavior.

3 Background and motivation

In this section, we briefly review the basic operations of DSR, investigate a potential attack, and measure the routing performance impact on DSR with a preliminary result.

3.1 Route discovery and maintenance

When a source node generates a Data packet to send, it first searches its routing table for the route to a destination node. If the route is not available, the source node initiates the route discovery procedure by broadcasting a route request packet (RREQ). Any intermediate node located between the source and destination nodes rebroadcasts the received RREQ by adding its node address in the packet header, if it does not have the route to the destination node. When the destination node receives the RREQ, it replies a route reply packet (RREP) back to the source node. Upon receiving the RREP, the source node sends a Data packet using the complete route of the destination node piggybacked in the packet header. If a link is broken during the transmission, a route error packet (RERR) is generated and forwarded back to the source node. Any node who overhears the RERR removes the route(s) containing the broken link from its routing table. Each node can quickly learn the routes of other nodes by aggressively overhearing on-flying packets and caching the piggybacked route information in its routing table. In DSR, overhearing does help in improving the routing performance but it may lead to a stale route or cache pollution problems [2]. The effect of overhearing in DSR is extensively analyzed in [25].

3.2 False destination of RREP

In [13], a cooperative bait detection scheme (CBDS) is proposed to detect both selective forwarding and blackhole attacks in DSR. When a source node receives an Alarm packet from a destination node for significant packet loss, it randomly selects one of its adjacent nodes as a bait destination node. Then the source node broadcasts a bait RREQ for enticing a potential malicious node to reply back a false RREP, containing a fake route of the bait destination node. If the source node receives the false RREP, it can identify the malicious node by using a reverse tracing technique. Note that the CBDS and its variant [26] do not consider the followings: (1) a source node could select a one-hop apart malicious node as a bait destination node. Thus, other malicious nodes may not reply to a bait RREQ by colluding with this selected malicious node; (2) a malicious node could overhear an on-flying RREQ and directly reply a false RREP piggybacked with a fake route of the destination node without including its node address. This can fail

the source node to identify the malicious node through a reverse tracing technique; and (3) prior approaches do not fully consider a case where malicious nodes ensure a certain level of packet dropping rate, leading packet delivery ratio (PDR) same or slightly higher than an Alarm threshold.

In Fig. 1, we measure the detection rate and number of dropped Data packets of CBDS by varying the packet rate (r_{pkt}) and percentage of malicious nodes in the network. In Fig. 1(a), as the percentage of malicious nodes increases, the detection rate decreases. This is because multiple malicious nodes can collude together and do not reply to a bait RREQ. Moreover, the number of dropped Data packets increases as the percentage of malicious nodes increases in Fig. 1(b). Since more malicious nodes can be actively involved in the routing operation, more Data packets are dropped.

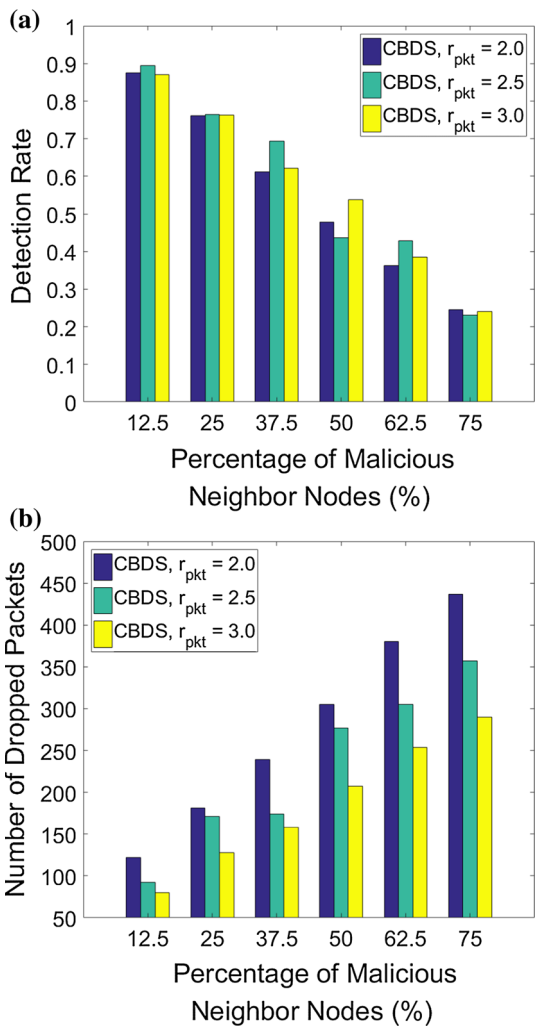


Fig. 1 The impact of the packet rate (r_{pkt}) and percentage of malicious nodes on the CBDS scheme. Here, we consider a network area 1000 × 1000 (m^2), where 100 nodes are uniformly distributed and move with a node speed, 8 (m/s). **a** Detection rate, **b** number of dropped packets

4 Proposed countermeasure

In this section, we first introduce an adversarial model and then propose an explore-based active detection scheme, called EBAD, to mitigate potential routing misbehaviors in MANETs running with DSR.

4.1 Adversarial model

We consider a set of homogeneous nodes that freely moves in a MANET, where each node is identified by its node address. In a network deployment phase, each node receives a public and private key-pair and the public key is globally available to other nodes [27]. An adversary is able to capture and compromise legitimate node, gain access to all stored information including public and private keys, and reprogram it to behave maliciously. The primary goal of the adversary is to disrupt the DSR protocol and interfere with on-going communication. A malicious node may selectively or strategically drop or forward any incoming packet on purpose. However, the malicious node will not blindly refuse to forward packets (i.e., blackhole attack) because its neighbor nodes may consider it as a failed node and select an alternative route. In order to mislead the network traffic, the malicious node may also overhear any on-flying packet and inject false route information in a RREP or modify the received packet. If a sender authenticates a packet with a light-weight digital signature [28], a receiver can easily verify whether the packet has been modified. In this paper, we primarily focus on a set of adversarial scenarios and its potential routing misbehaviors in DSR. One or multiple number of malicious nodes can falsely claim a fake shortest route to the destination node to be actively involved in the routing path and launch DoS attacks. We do not consider node capture attack [29], where an adversary can capture a legitimate node from the network as the first step for further different types of attacks. We also assume that the system is free of the general attacks such as sybil attack, collision or jamming attack, or wormhole attack.

4.2 Outline of the EBAD

The key idea of EBAD scheme is to utilize fake information to lure potential malicious nodes to reveal their routing misbehaviors. A source node broadcasts a route request packet with a fictitious destination node address to lure potential malicious nodes to reply a fake route reply packet. A malicious node may reply a route reply packet to

falsely claim that it has a route or the shortest route to the fictitious destination node to be involved in the routing operation and then launch denial-of-service (DoS) attack, e.g., selective forwarding attack. The malicious node in fact has no knowledge of whether the destination node address piggybacked in the route request packet exists in the network or not. When the source node receives the fake route reply packet, the routing misbehavior of the malicious nodes can be detected. When replying the fake route reply packet, the malicious node could attach faulty information to hide its identity and avoid detection, but this can also be detected by an integrated digital signature technique. A set of overall information flows of source and intermediate nodes is shown in Figs. 2, 3, and 4, respectively.

4.3 EBAD: explore-based active detection

The basic idea of the proposed scheme is that a source node broadcasts an exploring RREQ (eRREQ) piggybacked with a fictitious destination node address for luring potential malicious nodes to reply a fake RREP (fRREP) before initiating the route discovery procedure. Three major operations are followed.

Luring malicious nodes: When a source node is to initiate the route discovery procedure, malicious nodes might be located along the route to a destination node. The source node needs to decide whether to check the existence of malicious nodes in the route or not. We first deploy an exploring probability (P_e) that indicates how frequently the source node tries to check malicious nodes. P_e is adaptively adjusted depending on the detection frequency of malicious nodes, and its changes are observed in Sect. 5 (see Fig. 15).

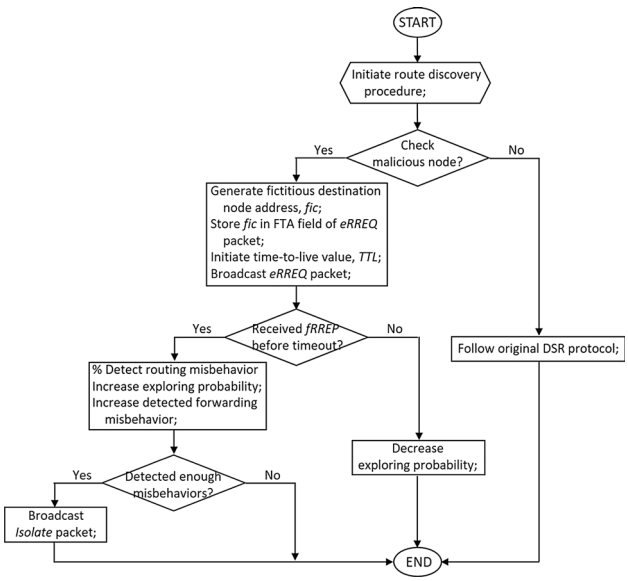


Fig. 2 An information flow of source node

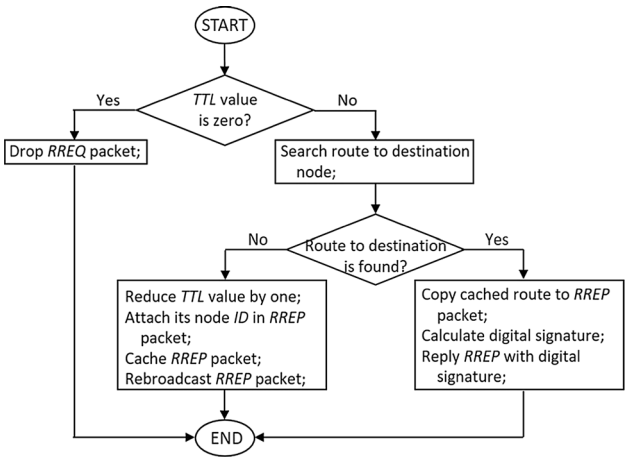


Fig. 3 An information flow of intermediate node that receives a RREQ

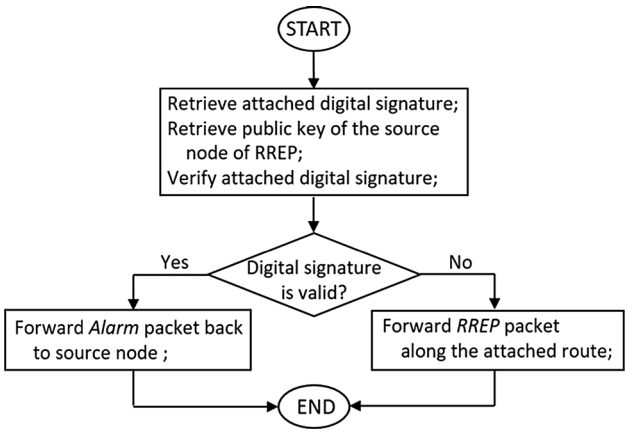


Fig. 4 An information flow of intermediate node that receives a RREP

If the source node generates a random number (e.g., rand[0,1]), which is less than or equal to P_e, it broadcasts an eRREQ piggybacked with a fictitious destination node address (*fic*). The source node can put the real or fictitious destination node address in a RREQ, which becomes an eRREQ if the fictitious destination node address is contained. The source node may create the unique *fic* derived either from its own media access control (MAC) address or a randomly generated fake MAC address [30]. Due to the constant size of MAC address (e.g., 48 bits), it is not guaranteed that every randomly generated fictitious destination node address is different from all real MAC addresses used in the network. However, the probability of generating a fake MAC address which is same as the existing address in the network will be extremely low and close to zero, because the 48-bit address space contains 2^{48} possible MAC addresses. Thus, we implicitly assume that a fictitious destination node, N_{fic}, does not exist in the network. If the source node decides not to broadcast an

eRREQ, it follows the original DSR protocol by initiating the route discovery procedure and broadcasts a traditional RREQ with the real destination node address. Here, each RREQ contains a time-to-live (*TTL*) value in terms of number of hops to limit packet propagation in the network. The *TTL* is decreased by one whenever RREQ is rebroadcasted. When a node receives a RREQ with $TTL = 0$, it does not rebroadcast but simply drops the RREQ. The format of modified RREQ is shown in Fig. 5, where the *fic* is stored in the FTA field and the DA field is set to the broadcast address [2]. The source node can either put the real or fictitious destination node address in the FTA field.

If a legitimate node receives the eRREQ, it always rebroadcasts the eRREQ because it does not have a route to the fictitious destination node in its routing table. However, a malicious node could reply a fRREP to falsely claim that it has a route or the shortest route to the fictitious destination node in order to be involved in the routing operation. Note that the fictitious destination node address is created by the source node based on a randomly generated fake MAC address and thus, only the source node knows that the destination node address contained in the eRREQ is fake. For malicious nodes, they have no knowledge of whether the destination node address contained in the eRREQ exists in the network. Since a new node may join the network or an existing node may leave the network without notice, the malicious node may have difficulty in determining whether the destination node address in the eRREQ is valid. Thus, the malicious node replies a fRREP to falsely claim that it has a route or the shortest route to a fictitious destination node. This will lead the malicious node to be involved in the future routing operation and have a chance to selectively or strategically drop or forward any incoming Data packet on purpose.

If an intermediate node cannot decrypt a received RREP piggybacked with a digital signature by using a public key of the source node of RREP, it prosecutes the RREP forwarding node for routing misbehavior by replying an Alarm packet back to the source node. If the source node receives a RREP packet corresponding to the broadcasted eRREQ, it prosecutes the source node of RREP for routing misbehavior. Otherwise, upon receiving the RREP, the source node sends a Data packet with the complete route

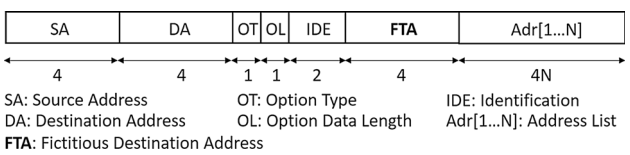


Fig. 5 The format of modified RREQ, where the source node can either put the real or fictitious destination node address in the FTA field. Here, the length is shown in byte

information piggybacked in the RREP. Note that if the malicious nodes try to drop the Data packet without forwarding on purpose, prior monitor-based [3, 7] or acknowledgment-based [6, 12] approaches can be deployed to detect potential forwarding misbehaviors. Major operations of luring malicious nodes in the network are summarized in Fig. 6.

Detecting and isolating malicious nodes: When a source node receives a RREP, it obtains a route in terms of a set of intermediate nodes located along the route to a destination node and a replier's node address. However, the source node may obtain a false route or wrongly identify the intermediate node that replies the RREP. This is because a malicious node can reply the RREP, falsely claim a fake route to the destination node, and modify the SA field to hide its identity. In order to detect any modification, we deploy a digital signature created by the source node of RREP based on a 1024-bit RSA digital signature technique [31, 32] and modify the traditional RREP accordingly. 1024-bit RSA technique has been widely used as a cryptographic primitive in wireless networks [23, 33], and has become a practical approach to provide required security services. The format of modified RREP is shown in Fig. 7. For example, when a node (e.g., N_i) replies a RREP (*rep*) claiming the route to the destination node, it puts its address into the SA field. N_i also can calculate a digital signature ($SG(i)$) using its private key (PrK_i) and put the $SG(i)$ into the Sig field, $md = H(rep)$ and $SG(i) = E_{PrKi}(md)$. Here, *md* is a fixed-length message digest calculated through a predefined hash function *H*, and $E_{PrKi}()$ denotes encryption with private key PrK_i. When a node (e.g., N_j) receives a RREP (rep') forwarded by another node (e.g.,

Notations:
- $RREQ[src, des, TTL, R]$: A RREQ with a source address (src), a destination address (des), time-to-live (TTL), and route record (R). If des contains fictitious destination address, $RREQ$ becomes $eRREQ$.
- $RREP[src, des, SG(src), R]$: A RREP with digital signature, $SG(src)$. src, des, and R are defined before.
- P_c: An exploring probability.

◇ When a source node, N_s, detects an event for a destination node, N_d:
  if $P_c <= rand[0, 1]$
    Generate a *fictitious* destination node address, fic;
    Broadcast an exploring RREQ, $RREQ[s, fic, TTL, R]$;
  else
    Broadcast a traditional RREQ, $RREQ[s, d, TTL, R]$;

◇ When an intermediate node N_i receives $RREQ[s, des, TTL, R]$:
  if $RREQ[TTL] == 0$
    Drop $RREQ[s, des, TTL, R]$;
  else
    Search the route to N_{des} in routing table;
    if the route to N_{des} is found
      Copy $RREQ[R]$ with cached route to $RREP[R]$;
      $md = H(RREP)$;
      $SG(i) = E_{PrK_i}(md)$;
      Reply $RREP[i, s, SG(i), R]$;
    else
      $RREQ[TTL]$ = $RREQ[TTL]$ - 1; Attach i in $RREQ[R]$;
      Cache $RREQ[R]$ and rebroadcast $RREQ[s, des, TTL, R]$;

Fig. 6 The pseudo code of luring malicious nodes in the network

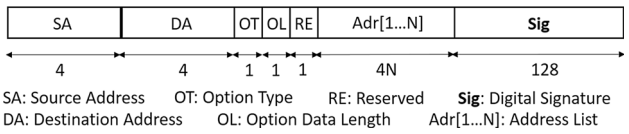


Fig. 7 The format of modified RREP, where a 1024-bit RSA digital signature is piggybacked in the Sig field. Here, the length is shown in byte

N_k), it verifies the attached digital signature using the public key of source node of rep' based on the SA field through $md' = H(rep')$ and $md = D_{PuKi}(SG(i))$. Suppose the source node of RREP is N_i. Here, PuK_i is the public key of N_i and $D_{PuKi}()$ denotes decryption with public key PuK_i. If md' based on rep' equals to md retrieved from attached digital signature, N_j chooses to forward the received RREP to the next node located in the packet header of RREP. Otherwise, N_j prosecutes N_k for a potential routing misbehavior and generates an *Alarm* packet and sends it back to the source node.

In Fig. 8, a source node (N_s) broadcasts an eRREQ piggybacked with a fictitious destination address (*fic*) to check the existence of malicious nodes before initiating the real route discovery procedure. When a malicious node (N_m) receives the eRREQ, it replies a fRREP to falsely claim that it has the route to N_{fic}. N_m puts its address (m) and the digital signature ($SG(m)$) calculated using its private key into the SA and Sig fields respectively in the fRREP, which will be replied back to N_s. If N_s receives the fRREP corresponding to prior eRREQ, it can detect the routing misbehavior because N_m claims the route to the fictitious destination node. Thus, N_m could attach a fake address and/or an invalid digital signature in the fRREP to hide its identity. However, this misbehavior can also be detected by the intermediate node located along the path. Each intermediate node verifies the attached digital signature of the received RREP with the public key of the source node of RREP. The major operations of detecting malicious nodes in the network are summarized in Fig. 9.

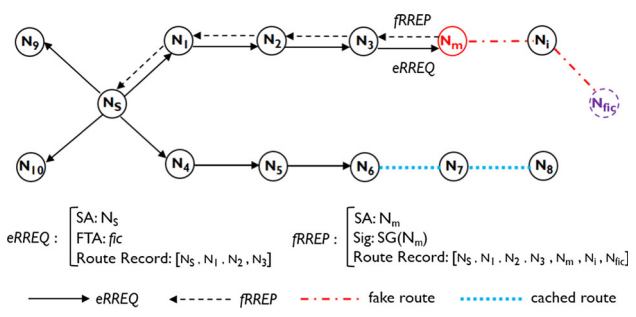


Fig. 8 A snapshot of the network, where a malicious node (N_m) falsely claims a route to the fictitious destination node (N_{fic}). N_m replies a fRREP corresponding to the eRREQ, originally propagated from a source node (N_s)

Notations:

- $RREQ[src, des, TTL, R]$, $RREP[src, des, SG(src), R]$, fic, P_e, T_{out}, δ, η, τ, md, md', H, PuK_x and PrK_x: Defined before.
- $Alarm[j]$: An Alarm packet reporting routing misbehavior of node N_j.
- $C_{mis}[j]$: The number of detected routing misbehaviors of node N_j.

◇ When source node N_s receives $RREP[m, s, SG(m), R]$ for *exploring* RREQ packet with fic:

```
if fic ∈ RREP[R] /* fic in route record RREP[R] */
    /* Increase exploring probability and detected routing misbehavior */
    P_e += δ; C_mis[m] += 1;
    if C_mis[m] >= τ
        Broadcast Isolate packet;
```

◇ When source node N_s doesn't receive $RREP[m, s, SG(m), R]$ for *exploring* RREQ packet with fic before T_{out}:

```
P_e −= η;
```

◇ When source node N_s receives $RREP[m, s, SG(m), R]$ for traditional RREQ packet:

```
Send Data packet through RREP[R];
```

◇ When normal node N_i receives $RREP[x, s, SG(y), R]$ from N_j:

```
md = D_PuK_x(SG(y));
md' = H(RREP);
if md ≠ md'
    Forward Alarm[j] back to N_s;
else
    Forward RREP[x, s, SG(y), R] through RREP[R];
```

Fig. 9 The pseudo code of detecting malicious node in the network

When a source node receives either an Alarm or a fRREP corresponding to eRREQ from one of the intermediate nodes, it increases the exploring probability and number of detected routing misbehaviors for the suspected node by δ and one, respectively. However, the source node decreases the exploring probability by η if it does not receive an Alarm or fRREP before a timeout period. When the number of detected routing misbehaviors reaches a threshold (τ), the source node broadcasts an *Isolate* packet to the network in order to prevent the suspected node from being involved in any routing operation. P_e is adaptively adjusted depending on the detection frequency of malicious nodes. τ is designed as a system parameter and can be configured depending on the urgency of removing malicious nodes in the network. For example, a communication critical network in battlefield or emergency rescue, τ is given a smaller value to quickly isolate and remove the adversary from the network. To balance the tradeoff between detection performance and resource utilization, τ can have a relatively large value in non-critical situation [34]. Here, both δ and η are system parameters and their impacts on the performance are observed in Sect. 5.

Preventing route cache pollution: Route caching via unconditional overhearing is one of the major features to improve routing performance in DSR. Whenever a node forwards or overhears a RREQ, RREP, or Data, it caches the route learned from the packet to its routing table. If a node forwards or overhears a RERR, it removes any route containing the broken link from its routing table. When a malicious node replies a fRREP, its fake route can contaminate the routing tables of intermediate nodes located along the path to a source node. In light of this, we deploy a

simple route expiry timer to purge fake routes from the routing table. Whenever a node uses a route to send a Data packet, it extends the expiration time of route. The rationale behind this approach is that when the source node receives the fRREP corresponding to prior eRREQ containing a fictitious destination node, it never uses the fake route learned from the fRREP to send a Data packet to the fictitious destination node. Thus, the fake route will eventually be expired and removed from the routing table. Note that the route cache pollution can be reduced by simply disabling the overhearing, but this can negatively affect the routing performance in terms of packet delivery ratio and packet delay [25]. In fact, the route cache pollution is unavoidable because of the changes of network topology due to the node mobility. In this paper, we do not consider an adaptive timeout period [35] based on the average route lifetime and the time between consecutive link breaks. Because it is hard to calculate the timeout period without non-negligible error in real time, but this is out of the scope of this paper.

4.4 Analysis of the EBAD

We further analyze the proposed approach in terms of detection rate. When a source node receives an Alarm or the fRREP corresponding to an eRREQ from one of intermediate nodes, it can detect the routing misbehavior of malicious nodes. Suppose a network size is $X \times Y$ (m^2), where N nodes are uniformly distributed, and a packet loss rate is ϕ due to the channel error or node mobility. With an exploring probability, P_e, the source node broadcasts an eRREQ to check the existence of any malicious node. We assume that N_m is the first malicious node that receives the eRREQ and replies the fRREP. Let P_{detect} be a detection rate, which is the sum of probabilities of receiving the fRREP (P_{frep}) or the Alarm (P_{alarm}). Then P_{detect} is expressed as,

$$P_{detect} = P_{frep} + P_{alarm} \tag{1}$$

Here, the average number of hops between the source node and N_m, h, is approximated according to [6] and it is expressed as,

$$h \approx \frac{d}{\ell} \approx \frac{\sqrt{X^2 + Y^2}}{2\ell} \approx \frac{(2\xi + 1) \cdot \sqrt{X^2 + Y^2}}{4\xi R} \tag{2}$$

Here, ℓ and d are the average progress of each hop and average distance between the source node and N_m, respectively. R is the communication range of each node. ξ is the average number of nodes located within R and it is expressed as,

$$\xi = \frac{N}{X * R} \cdot \pi R^2 \tag{3}$$

First, P_{frep} is expressed as,

$$P_{freq} = P_{mr} \cdot P_{sr}, \tag{4}$$

where $P_{mr} = P_e \cdot (1 - \phi)^h$ and $P_{sr} = (1 - \phi)^h$. Here, P_{mr} is a probability of N_m receiving the eRREQ through h multi-hop relays. P_{sr} is a probability of the source node receiving the fRREP. Second, P_{alarm} is expressed as,

$$P_{alarm} = P_{mr} \cdot P_{sa} \tag{5}$$

where $P_{sa} = (1 - \phi) \cdot (1 - \phi)^{h-1}$. Here, P_{sa} is a probability that an Alarm is generated by the node located in front of N_m and forwarded back to the source node. Finally, P_{detect} is expressed as,

$$\begin{aligned} P_{detect} &= 2 \cdot P_e \cdot (1 - \varphi)^{2h} \\ &= 2 \cdot P_e \cdot (1 - \varphi) \frac{(2\xi + 1) \cdot \sqrt{X^2 + Y^2}}{4\xi R} \end{aligned} \tag{6}$$

In Fig. 10, we show a numerical result of the detection rate against the exploring probability. Here, 100 nodes are uniformly distributed in a 1000 × 1000 (m^2) network area, where the communication range of each node and the packet loss rate are 250 (m) and 5%, respectively. The detection rate increases linearly as the exploring probability increases, because the source node has more chances to perform the proposed scheme by frequently broadcasting an eRREQ and detects more routing misbehaviors. Since the proposed analytical model is intended only to estimate the performance trend of detection rate, we conduct detail performance evaluation through extensive simulation in the following section.

5 Performance evaluation

5.1 Simulation testbed

We conduct extensive simulation experiments using the OMNeT++ [14] for performance evaluation and analysis.

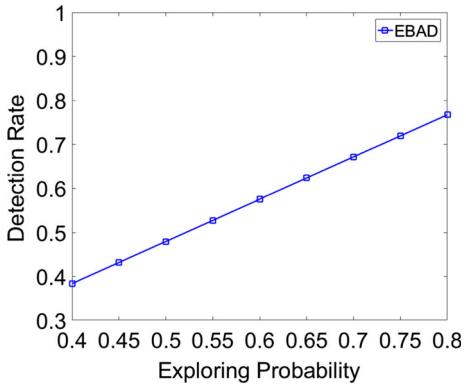


Fig. 10 The detection rate against the exploring probability

A 1000 × 1000 (m^2) rectangular network area is considered, where 100 nodes are uniformly distributed. Ten malicious nodes out of 100 nodes are randomly located in the network and selectively drop any incoming Data packet with a drop rate, 50%. The radio transmission range is assumed to be 250 (m) and the two-way ground propagation channel is assumed with a data rate of 2 Mbps. The source nodes generate a constant bit rate (CBR) traffic at the packet injection rate of 0.5–3.0 packet/s and each packet size is 512 Bytes. The random waypoint mobility model [2] is deployed in the network, where each node travels toward a randomly selected destination in the network with a speed (s) between 0 and 10 (m/s). Upon reaching the destination, each node pauses for 10–40 s, travels toward another randomly selected destination, and repeats travel and pause operations. The total simulation time is 1000 s, and each simulation scenario is repeated 10 times to obtain steady state performance metrics.

5.2 Performance comparison

In this paper, we measure six performance metrics including detection rate, energy consumption, packet delivery ratio (PDR), node behavior distribution, exploring probability, and statistics of Data packets by changing key simulation parameters, including packet injection rate (r_{pkt}), weights (δ and η) of exploring probability, and node speed.

- *Detection rate*: Detection rate is computed as the ratio of the number of received fRREP and Alarm to the total number of generated eRREQ, aiming to compare the detection efficiency of the proposed EBAD, CBDS, and 2ACK.
- *Energy consumption*: Energy consumption is measured based on the number of forwarded and received packets [36], and it is used to compare energy efficiency and consumption of the proposed EBAD, DSR, CBDS, and 2ACK.
- *Packet delivery ratio* (*PDR*): PDR is computed as the ratio of the number of received Data packets to the total number of generated Data packets, showing the performance resiliency of the proposed EBAD, DSR, CBDS, and 2ACK in the adversary scenarios.
- *Node behavior distribution*: The node behaviors of forwarding and dropping Data packets are recorded for entire simulation time, indicating how quickly the malicious nodes can be isolated and removed from the network by the proposed EBAD and CBDS.
- *Exploring probability*: The frequency of checking the existence of malicious nodes by source node is recorded for entire simulation time, showing how the exploring probability is adjusted depending on the detection of malicious nodes.
- *Statistics of Data packets*: The changes of the number of generated, delivered, and dropped Data packets are recorded, indicating the performance resiliency of the proposed EBAD, CBDS, and DSR.

We compare the performance of proposed scheme, EBAD, with the CBDS [13] and 2ACK [6]. The rationale behind choosing the CBDS and 2ACK is that they similarly use a fake information and extra control packet (i.e., ACK packet) to detect the routing misbehaviors in MANETs. Although many variants of the CBDS and 2ACK have been suggested, we focus on their major operations for comparison and briefly describe them below:

- *CBDS*: A destination node replies an Alarm packet back to a source node if the observed PDR is lower than a threshold value. After receiving the Alarm packet, the source node broadcasts a RREQ with a randomly selected one-hop neighbor node's address as the destination address to entice a malicious node to send back a RREP. Here, the Alarm threshold values are set between 0.85 and 0.9.
- *2ACK*: After receiving a Data packet, each intermediate node located along the forwarding path generates an ACK packet based on an acknowledgment ratio, and forwards it to the two-hop neighbor node located in the opposite direction. If the intermediate node cannot receive the ACK packet corresponding to the previously forwarded Data packet before a timeout period, the routing misbehavior is detected. Here, the acknowledgment ratio (R_{ack}) is configured to 1.0 and 0.5.

In addition, original DSR [2] without detection scheme is used as the lower bound of performance in PDR and energy consumption for performance comparison.

Detection rate: We first measure detection rate by changing the packet injection rate and node speed in Fig. 11. In Fig. 11(a), the detection rate of the CBDS increases as r_{pkt} increases. This is because PDR is more sensitive to packet loss when less number of Data packets are generated at a source node, leading to drop PDR below the threshold value. Then a destination node frequently replies an Alarm packet and the source node has more chances to broadcast a bait RREQ to detect more routing misbehaviors. In the 2ACK, less number of Data packets are dropped by malicious nodes as r_{pkt} increases. However, these routing misbehaviors can be detected and detection rate increases. In addition, detection rate of $R_{ack} = 1.0$ is larger than that of $R_{ack} = 0.5$, this is because more ACK packets are generated with $R_{ack} = 1.0$, and more routing misbehaviors can be detected. The EBAD shows higher detection rate than that of the CBDS and 2ACK. This is

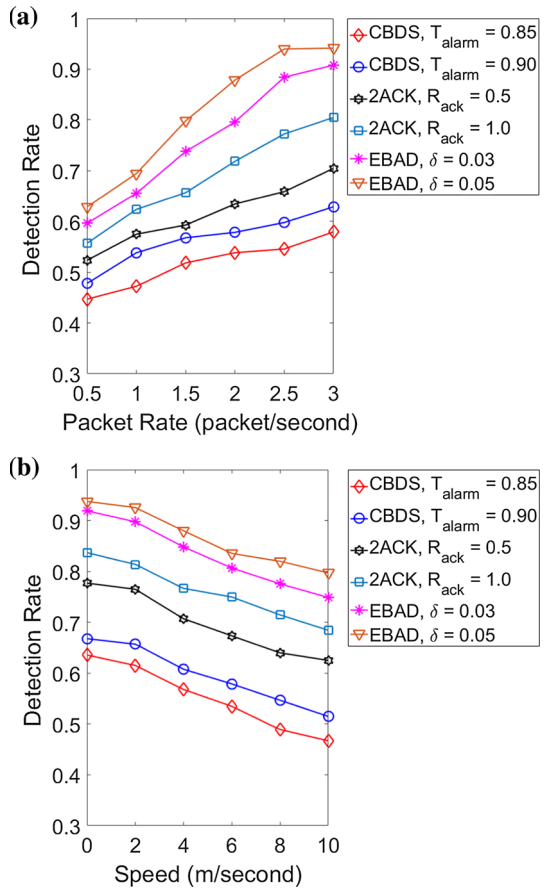


Fig. 11 The detection rate against the packet injection rate and node speed. **a** $s = 8$ m/s, **b** $r_{pkt} = 2.0$ packet/s

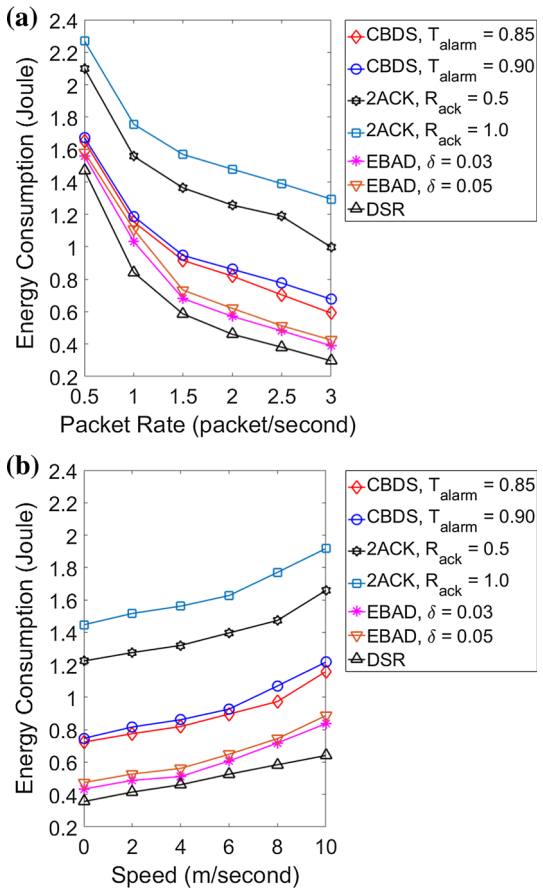


Fig. 12 The energy consumption against the packet injection rate and node speed. **a** $s = 8$ m/s, **b** $r_{pkt} = 2.0$ packet/s

because the source node under the CBDS could select a one-hop apart malicious node as a bait destination address, and the malicious nodes may not reply to a bait RREQ. In the 2ACK, two consecutively located malicious nodes can cooperate to reply an ACK packet after dropping a Data packet, but this cooperative routing misbehavior cannot easily be detected. The EBAD uses a fictitious destination address to check potential malicious node before initiating the route discovery procedure, and thus more routing misbehaviors can be detected. In Fig. 11(b), overall detection rates decrease because links are frequently broken as the speed increases, but the EBAD still shows higher detection rate compared to that of the CBDS and 2ACK.

Energy consumption: We measure energy consumption by changing the packet injection rate and node speed in Fig. 12. In Fig. 12(a), DSR shows the lowest energy consumption because it does not include any additional operation for detection. The 2ACK with different acknowledgment ratios shows the highest energy consumption because of large amount of ACK packets traversed along the forwarding path. The CBDS shows more energy consumption than that of the EBAD because of a large number of control packets traversed along the forwarding path for detection. The EBAD with different δ shows higher and lower energy consumption than that of DSR and CBDS respectively because eRREQ is added for detection. To explore the existence of malicious nodes and detect the routing misbehaviors, extra control packets are required in the proposed EBAD. However, since the exploring probability can be adaptively adjusted depending on the detection frequency of the routing misbehaviors, the malicious nodes can be quickly isolated and removed from the network. After removing the malicious nodes, no extra control packets will be generated for the detection of routing misbehavior in the network, and it follows the original DSR protocol. Therefore, the proposed EBAD will not create significant network congestion and energy consumption problems. In Fig. 12(b), energy consumption of the CBDS, 2ACK and EBAD increases as the node speed increases. Since high node mobility incurs more links broken, more control packets are required to update the routes and thus, more energy consumption is observed. However, the EBAD still shows lower energy consumption compared to that of the CBDS and 2ACK because less

number of control packets (i.e., eRREQ and Alarm) are added for detection.

Packet delivery ratio: We measure the PDR by changing the packet injection rate and node speed in Fig. 13. In Fig. 13(a), DSR is not sensitive to the packet injection rate but the packet drop rate (i.e., 50%) and thus, the PDR is fluctuating around 40%. This is because a malicious node randomly drops any received Data packet. The Data packet also could be lost during the transmission because of the node mobility. The CBDS shows higher PDR than that of DSR as r_{pkt} increases because the CBDS sends a bait RREQ to detect the routing misbehavior of malicious nodes. Thus, more Data packets can be delivered to the destination node. Higher PDR can also be achieved with higher Alarm threshold value 0.9. This is because the destination node frequently replies an Alarm packet back to the source node. Then the source node has more chances to detect the routing misbehaviors by broadcasting bait RREQ. The 2ACK shows higher PDR than that of the CBDS with different R_{ack} because the malicious nodes can be quickly isolated and lower number of Data packets is dropped. The EBAD with $\delta = 0.03$ and 0.05 shows the highest PDR compared to that of the CBDS, 2ACK, and DSR. This is because the source node actively sends an eRREQ to lure the malicious node to reply a fRREP. More routing misbehaviors can be detected and more malicious nodes can be isolated from the network. In Fig. 13(b), overall PDRs decrease as the node speed increases. However, the EBAD still shows the best performance compared to that of the CBDS, 2ACK, and DSR. Since the source node can actively broadcast an eRREQ in the EBAD, potential malicious nodes can be quickly detected and isolated from the network, leading more Data packets delivered to the destination node.

Node behavior: We observe and record a series of node behaviors running under the CBDS and EBAD for entire simulation time in Fig. 14. Here, 1 and -1 indicate the behaviors of forwarding and dropping Data packets, respectively. In Fig. 14(a), a series of dropping behaviors is observed from beginning to approximately 760 s. Then only forwarding behaviors are observed, indicating that the CBDS detects routing misbehaviors and isolates malicious

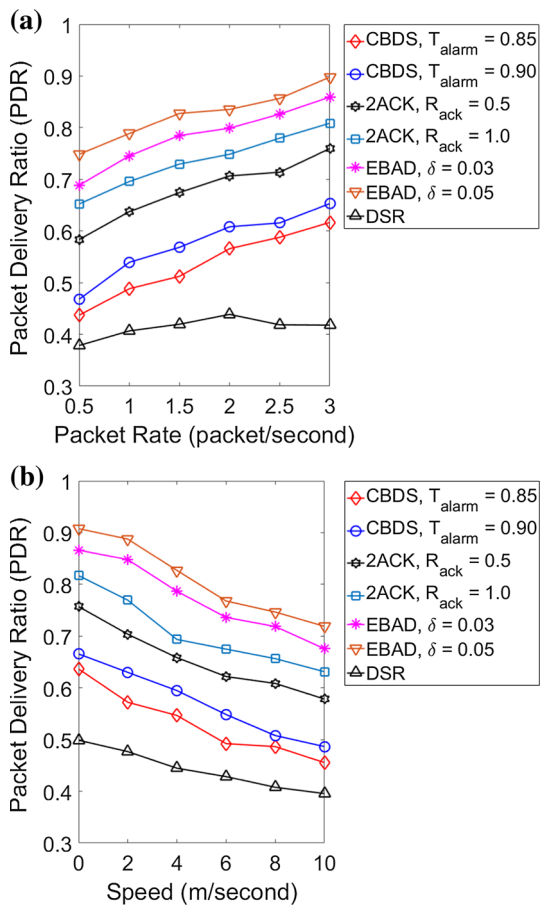


Fig. 13 The PDR against the packet injection rate and node speed. **a** $s = 8$ m/s, **b** $r_{pkt} = 2.0$ packet/s

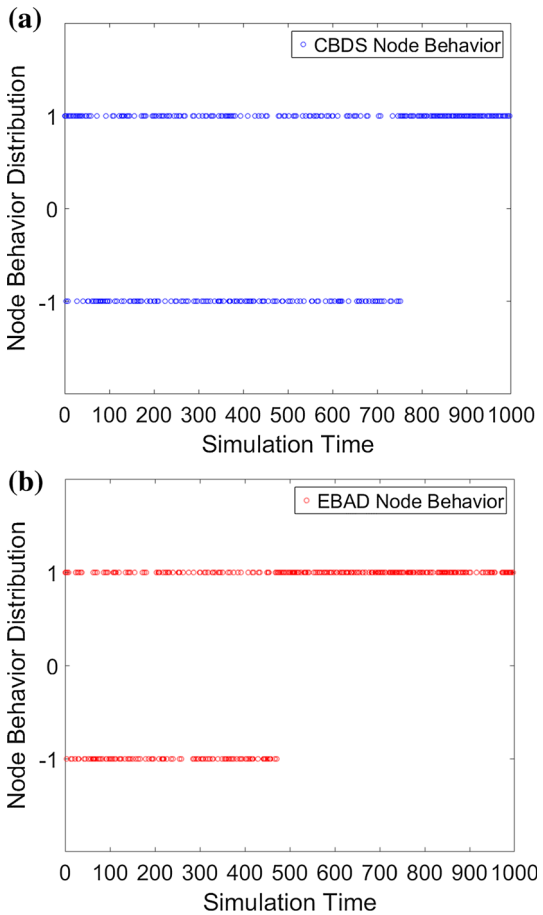


Fig. 14 The series of node behaviors during the simulation time. **a** CBDS, **b** EBAD

nodes from the network around 760 s. In Fig. 14(b), the EBAD shows earlier termination of dropping behaviors around 470 s. Since the source node can actively send an eRREQ to detect routing misbehaviors, malicious nodes can be quickly isolated from the network. This result indicates that the EBAD can provide lower detection latency compared to that of the CBDS.

Change of exploring probability: We observe the change of exploring probability against different increment and decrement weights (i.e., $\delta = 0.01$, 0.03, or 0.05, $\eta = 0.01$ or 0.02) in Fig. 15. Whenever a source node detects a routing misbehavior by receiving an Alarm or a fRREP corresponding to an eRREQ, it increases the exploring probability by δ. If the source node does not receive the Alarm or fRREP before a timeout period, it decreases the exploring probability by η. For example, the exploring probability with $\delta = 0.05$ reaches to 1.0 by approximately 251 s. In this paper, we also consider a case without any malicious node, denoted as 0-M, to see the change of exploring probability. The exploring probability with $\eta = 0.01$ and 0.02 reaches to 0 around 493 and 250 s, respectively. This implies that the EBAD is not operated anymore but the traditional DSR is operated in the network.

Statistics of Data packet: Finally, we measure the number of generated, delivered, and dropped Data packets by changing the packet injection rate in the DSR, CBDS and EBAD, respectively in Fig. 16(a)–(c). In Fig. 16(a), a small number of Data packets are delivered to the destination but most Data packets are dropped by malicious nodes in DSR. The CBDS generates lower number of Data packets compared to that of DSR, but higher number of Data packets are delivered to the destination in Fig. 16(b). In Fig. 16(c), we can observe that higher number of Data delivered but lower number of dropped Data packets in the

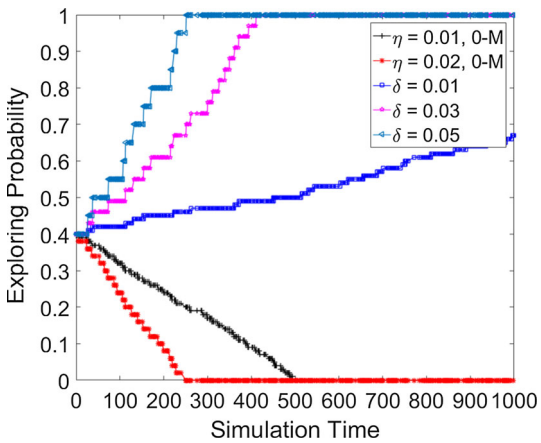


Fig. 15 The change of exploring probability against δ and η

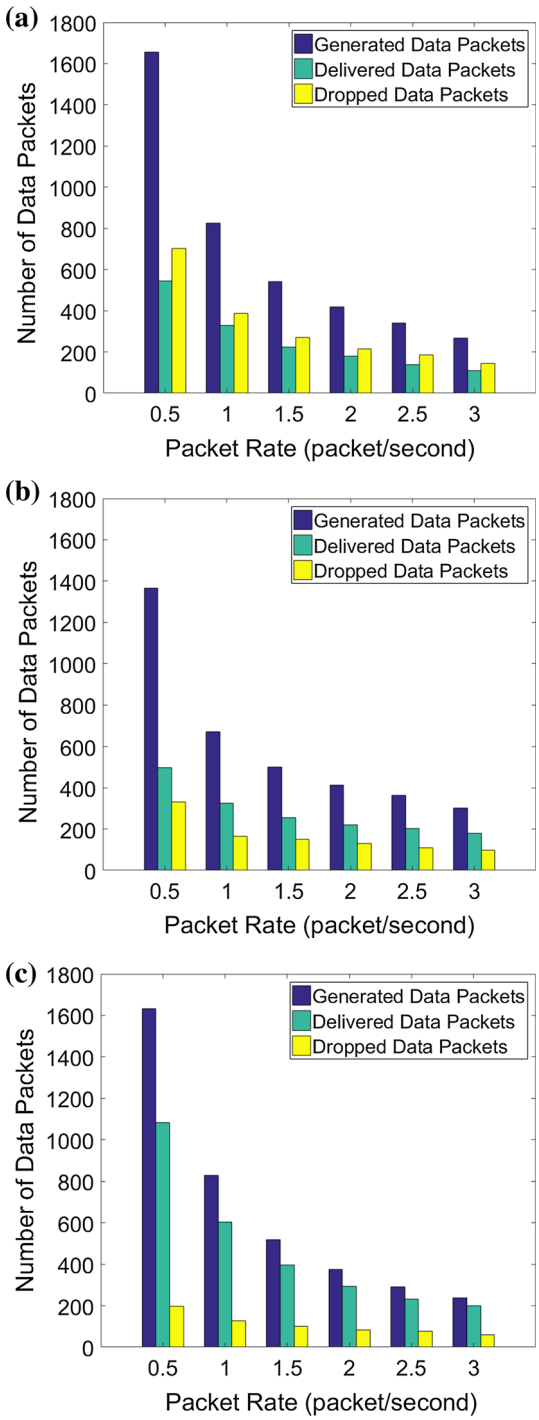


Fig. 16 The change of the number of generated, delivered, and dropped Data packets against the packet injection rate. **a** DSR, **b** CBDS, **c** EBAD

EBAD. This is because the EBAD can quickly detect more routing misbehaviors by actively sending an eRREQ and quickly isolate potential malicious nodes from the network. Thus, more Data packets can be delivered.

6 Discussion

In this section, we further analyze the performance of the EBAD with both CBDS and 2ACK and explore design issues and possible extensions to see the full potential of our approach.

6.1 Analysis of performance evaluation

We compare the EBAD with the existing schemes, CBDS and 2ACK, respectively.

EBAD vs. CBDS: It is known that the detection process of CBDS could fail if multiple malicious nodes collude together in the network. For example, a source node could select a one-hop malicious node as a bait destination node that can collude with other malicious nodes not to reply to a bait RREQ. Moreover, the malicious node could directly reply a false RREP piggybacked with a fake route without including its node address. This can cause the source node to fail to identify the malicious node through the proposed reverse tracing technique. In the EBAD, however, a source node generates a fictitious destination node that does not actually exist in the network. The malicious node should reply a fake route reply packet to be involved in the routing operation, because the malicious node has no knowledge of whether the fictitious destination node exists or not. The EBAD is also integrated with the digital signature technique to detect any faulty information. As shown in Fig. 11, the detection rate of EBAD is obviously larger than that of the CBDS.

EBAD vs. 2ACK: In the 2ACK, each intermediate node located along the forwarding path generates an ACK packet and forwards it to a two-hop neighbor node in the opposite direction of the data traffic to detect routing misbehavior. Thus, a large number of control packets can be generated and forwarded by intermediate nodes, causing a non-neglectable amount of energy consumption. In the EBAD, an exploring probability is deployed to adaptively adjust the detection frequency of malicious nodes. Since the malicious nodes can be quickly isolated and removed from the network, the lower number of control packets are generated for the detection of routing misbehaviors, decreasing overall energy consumption compared to that of the 2ACK as shown in Fig. 12.

6.2 Potential enhancements

In this paper, the EBAD is seamlessly integrated with DSR, whose routing performance is highly dependent on active overhearing. In [25], we showed the effect of overhearing in terms of routing performance, such as PDR, packet delay, and number of packets transmitted and overheard. With little or no overhearing, the routing performance of DSR reduces significantly. The EBAD is neither designed to incur overhearing nor utilize the overhearing of DSR to conduct detection operations of malicious nodes.

The EBAD is designed based on an implicit assumption that the malicious nodes are not always be located along the shortest path to the destination. To be involved in the future routing operations and launch DoS attacks, the malicious node replies a fake RREP to falsely claim that it has a route or the shortest route to the destination node. However, the EBAD is not originally designed to deal with an adversarial scenario that the malicious node is located in the shortest path to the destination node and then drop the received Data packet. In this case, either monitor-based [3, 7] or acknowledgement-based [6, 12] detection approach can also be deployed to detect potential routing misbehaviors. Moreover, the EBAD has been applied to DSR, which is a reactive routing protocol just like Ad hoc On-Demand Distance Vector (AODV) routing [37]. We plan to apply the EBAD to a proactive routing protocol, for example Destination-Sequenced Distance Vector (DSDV) [38] and compare the routing performance. We also plan to investigate the impact of level of overhearing in DSR, such as no overhearing and randomized overhearing [25], on the EBAD.

7 Concluding remarks

In this paper, we proposed an explore-based active detection, called EBAD, to mitigate the routing misbehaviors in MANETs running with DSR. In the EBAD, an exploring RREQ packet piggybacked with a fictitious destination node address is used to lure potential malicious nodes to reply a fake RREP packet. The malicious nodes can be detected and isolated from being involved in the routing through a digital signature technique and an Alarm packet, respectively. A route expiry timer is also proposed to reduce the effect of route cache pollution. In addition, a simple analytical model of the EBAD and its numerical result in terms of detection rate are presented. Extensive simulation results indicate that the proposed countermeasure achieves better performance in terms of detection rate, energy consumption, PDR, and detection latency compared to the CBDS and 2ACK.

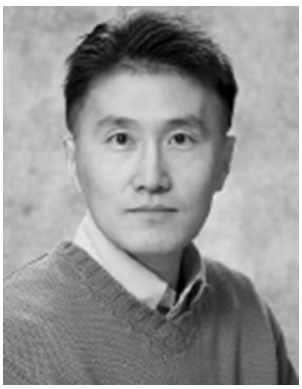
Acknowledgements This research was supported in part by Startup grant in Division of Computer Science at Marshall University.

References

1. Loo, J., Mauri, J. L., & Ortiz, J. H. (2016). *Mobile ad hoc networks: Current status and future trends*. Boca Raton: CRC Press.

2. Johnson, D. B., & Maltz, D. A. (1996). Dynamic source routing in ad hoc wireless networks. In T. Imielinski, & H. F. Korth (Eds.), *Mobile computing* (pp. 153–181). Boston, MA: Springer.
3. Marti, S., Giuli, T. J., Lai, K., & Baker, M. (2000). Mitigating routing misbehavior in mobile ad hoc networks. In *Proceedings of ACM MOBICOM*, pp. 255–265.
4. Yu, B., & Xiao, B. (2006). Detecting selective forwarding attacks in wireless sensor networks. In *Proceedings of IEEE IPDPS*, pp. 1–8.
5. Xiao, B., Yu, B., & Gao, C. (2007). CHEMAS: Identify suspect nodes in selective forwarding attacks. *Journal of Parallel and Distributed Computing, 67*(11), 1218–1230.
6. Liu, K., Deng, J., Varshney, P. K., & Balakrishnan, K. (2007). An acknowledgment-based approach for the detection of routing misbehavior in MANETs. *IEEE Transactions on Mobile Computing, 6*(5), 536–550.
7. Shila, D. M., Yu, C., & Anjali, T. (2010). Mitigating selective forwarding attacks with a channel-aware approach in WMNs. *IEEE Transactions on Wireless Communications, 9*(5), 1661–1675.
8. Liu, Q., Yin, J., Leung, V., & Cai, Z. (2013). FADE: Forwarding assessment based detection of collaborative grey hole attacks in WMNs. *IEEE Transactions on Wireless Communications, 12*(10), 5124–5137.
9. Chae, Y., DiPippo, L. C., & Sun, Y. L. (2015). Trust management for defending on-off attacks. *IEEE Transactions on Parallel and Distributed Systems, 26*(4), 1178–1191.
10. Ren, J., Zhang, Y., Zhang, K., & Shen, X. S. (2014). Exploiting channel-aware reputation system against selective forwarding attacks in WSNs. In *Proceedings of IEEE GLOBECOM*, pp. 330–335.
11. Ren, J., Zhang, Y., Zhang, K., & Shen, X. (2016). Adaptive and channel-aware detection of selective forwarding attacks in wireless sensor networks. *IEEE Transactions on Wireless Communications, 15*(5), 3718–3731.
12. Pu, C., & Lim, S. (2016). A light-weight countermeasure to forwarding misbehavior in wireless sensor networks: Design, analysis, and evaluation. *IEEE Systems Journal*. https://doi.org/10.1109/JSYST.2016.2535730.
13. Chang, J.-M., Tsou, P.-C., Woungang, I., Chao, H.-C., & Lai, C.-F. (2015). Defending against collaborative attacks by malicious nodes in MANETs: A cooperative bait detection approach. *IEEE Systems Journal, 9*(1), 65–75.
14. A. Varga. (2014). OMNeT++. http://www.omnetpp.org/.
15. Lim, S., & Lauren, H. (2015). Hop-by-hop cooperative detection of selective forwarding attacks in energy harvesting wireless sensor networks. In *Proceedings of IEEE ICNC*, pp. 315–319.
16. Hu, Y.-C., Perrig, A., & Johnson, D. B. (2005). Ariadne: A secure on-demand routing protocol for ad hoc networks. *Wireless Networks, 11*(1–2), 21–38.
17. Perrig, A., Canetti, R., Tygar, J. D., & Song, D. (2002). The TESLA broadcast authentication protocol. *RSA CryptoBytes, 5*(2), 2–13.
18. Dong, Y., Chim, T. W., Li, V. O., Yiu, S.-M., & Hui, C. (2009). ARMR: Anonymous routing protocol with multiple routes for communications in mobile ad hoc networks. *Ad Hoc Networks, 7*(8), 1536–1550.
19. Li, X., Li, H., Ma, J., & Zhang, W. (2009). An efficient anonymous routing protocol for mobile ad hoc networks. In *Fifth international conference on information assurance and security, 2009 (IAS'09)*, vol. 2. IEEE, pp. 287–290.
20. Song, R., Korba, L., & Yee, G. (2005). AnonDSR: Efficient anonymous dynamic source routing for mobile ad-hoc networks. In *Proceedings of the 3rd ACM workshop on security of ad hoc and sensor networks*. ACM, pp. 33–42.
21. Pu, C., & Lim, S. (2015). Spy vs. spy: Camouflage-based active detection in energy harvesting motivated networks. In *Proceedings of IEEE MILCOM*, pp. 903–908.
22. Midi, D., & Bertino, E. (2016). Node or link? Fine-grained analysis of packet-loss attacks in wireless sensor networks. *ACM Transactions on Sensor Networks, 12*(2), 8.
23. Zhang, Y., Lazos, L., & Kozma, W. (2016). AMD: Audit-based misbehavior detection in wireless ad hoc networks. *IEEE Transactions on Mobile Computing, 15*(8), 1893–1907.
24. Stehlik, M., Matyas, V., & Stetsko, A. (2016). Towards better selective forwarding and delay attacks detection in wireless sensor networks. In *Proceedings of IEEE ICNSC*, pp. 1–6.
25. Lim, S., Yu, C., & Das, C. R. (2009). RandomCast: An energy efficient communication scheme for mobile ad hoc networks. *IEEE Transactions on Mobile Computing, 8*(3), 351–369.
26. Haghighi, A., Mizanian, K., & Mirjalily, G. (2015). Modified CBDS for defending against collaborative attacks by malicious nodes in MANETs. In *2nd international conference on KBEI*. IEEE, pp. 902–907.
27. Kim, J., & Tsudik, G. (2009). SRDP: Secure route discovery for dynamic source routing in MANETs. *Ad Hoc Networks, 7*(6), 1097–1109.
28. Stallings, W. (2013). *Cryptography and network security—Principles and practices* (6th ed.). Upper Saddle River: Prentice Hall.
29. Conti, M., Pietro Di, R., Mancini, L., & Mei, A. (2008). Emergent properties: Detection of the node-capture attack in mobile wireless sensor networks. In *Proceedings of ACM wireless network security*, pp. 214–219.
30. Reibel, J. (2002). An IP address configuration algorithm for zeroconf. mobile multi-hop ad-hoc networks. In *Proceedings of the international workshop on broadband wireless ad-hoc networks and services*. Citeseer, Sophia Antipolis.
31. Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM, 21*(2), 120–126.
32. Hu, Y.-C., Perrig, A., & Johnson, D. B. (2003). Packet leashes: A defense against wormhole attacks in wireless networks. In *Proceedings of IEEE INFOCOM*, pp. 1976–1986.
33. Chatterjee, P., Ghosh, U., Sengupta, I., & Ghosh, S. (2014). A trust enhanced secure clustering framework for wireless ad hoc networks. *Wireless Networks, 20*(7), 1669–1684.
34. Li, X., Lu, R., Liang, X., & Shen, X. (2011). Side channel monitoring: Packet drop attack detection in wireless ad hoc networks. In *Proceedings of IEEE ICC*, pp. 1–5.
35. Marina, M. K., & Das, S. R. (2001). Performance of route caching strategies in dynamic source routing. In *International workshop on wireless networks and mobile computing (WNMC)*, pp. 425–432.
36. Tang, X., & Xu, J. (2006). Extending network lifetime for precision-constrained data aggregation in wireless sensor networks. In *Proceedings of IEEE INFOCOM*, pp. 1–12.
37. Perkins, C., & Belding-Royer, E. (1999). Ad hoc on-demand distance vector routing. In *Proceedings of WMCSA*, pp. 90–100.
38. Perkins, C., & Bhagwat, P. (1994). Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *Proceedings of SIGCOMM*, pp. 234–244.

Cong Pu received his B.S. degree in Computer Science and Technology from Zhengzhou University (China) in 2009. He earned his M.S. and Ph.D. degrees in Computer Science from Texas Tech University in 2013 and 2016, respectively. Currently, he is an Assistant Professor in the Weisberg Division of Computer Science, Marshall University (MU). Before joining MU, he was an Instructor in the Department of Computer Science at Texas Tech University from 2014 and 2016. His research interests are in the areas of Cybersecurity, Wireless Networks and Mobile Computing, Energy Harvesting Motivated Networks, Mobile Ad Hoc Networks, Low Power and Lossy Networks, and Evacuation Assisting Vehicular Networks. He was the reviewer of several IEEE journals and serves as Technical Program Committee in SPACOMM 2018. He is a member of the IEEE. He received 2015 Helen Devitt Jones Excellence in Graduate Teaching Award at Texas Tech University. He was the Winner of 2017 Design for Delight (D4D) Innovation Challenge Competition as Faculty Coach (Marshall University and Intuit Inc.).



Sunho Lim received the B.S. degree (summa cum laude) in Computer Science and the M.S. degree in Computer Engineering from Hankuk Aviation University (a.k.a. Korea Aerospace University), Goyang, Korea, in 1996 and 1998, respectively, and the Ph.D. degree in Computer Science and Engineering from Pennsylvania State University, State College, PA, USA, in 2005. He is currently an Assistant Professor with the Department of Computer Science, Texas Tech University (TTU), Lubbock, TX, USA. Before joining TTU, he was an Assistant Professor with the Department of Electrical Engineering and Computer Science, South Dakota State University, Brookings, SD, USA, from 2005 to 2009. His research interests include areas of Cybersecurity, Mobile Data Management and Privacy, and Wireless Networks and Mobile Computing. Dr. Lim was a Guest Editor of special issue on dependability and security for Wireless Ad Hoc and Senor Networks and their applications in Int'l Journal of Distributed Sensor Networks, and has served on the NSF proposal panel and program committees of many renowned conferences. He is leading the T^2WISTOR: TTU Wireless Mobile Networking Laboratory. He was the recipient of Texas Tech Alumni Association New Faculty Award and Air Force and Navy Summer Faculty Fellowship. He is a senior member of IEEE.

Jinseok Chae received his B.S., M.S., and Ph.D. degrees in Computer Engineering from Seoul National University, Korea in 1990, 1992, and 1998, respectively. He is currently a professor in the Department of Computer Science and Engineering at Incheon National University, Korea. Prior to joining Incheon National University, he worked in the Engineering Laboratory at Seoul National University and the Korea Research Information Center, Korea. He was a visiting scholar in the Department of Computer Science at California State University San Bernardino, California, USA in 2006 and in the Department of Computer Science at Texas Tech University, Texas, USA from 2015 to 2017. He served as a dean of admissions and student affairs at Incheon National University from 2012 to 2014 and an editor-in-chief of Journal of KIISE: Computing Practices and Letters from 2011 to 2014. His research interests include internet software, web technology and mobile computing. He is a member of IEEE since 1992.

Byungkwan Jung received his B.S. degree in Accounting – Taxation from Kyunghee University, Seoul, Republic of Korea and M.S. degree in the Department of Computer Science from South Dakota State University, Brookings in 2010 and 2014, respectively. He is currently a Ph.D. candidate in the Department of Computer Science at Texas Tech University. His research interests include wireless networks and mobile computing, Cybersecurity, image processing, and machine learning.