IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 18, 2019, accepted July 10, 2019, date of publication July 15, 2019, date of current version August 1, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2928804

BP-AODV: Blackhole Protected AODV Routing Protocol for MANETs Based on Chaotic Map

ALY M. EL-SEMARY[1,2], (Member, IEEE), AND HOSSAM DIAB[3,4]
[1]Computer Engineering Department, College of Computer Science and Engineering, Taibah University, Madinah 344, Saudi Arabia
[2]Systems and Computer Engineering Department, Faculty of Engineering, Al-Azhar University, Cairo 11651, Egypt
[3]Computer Science Department, College of Computer Science and Engineering, Taibah University, Madinah 344, Saudi Arabia
[4]Math and Computer Science Department, Faculty of Science, Menoufia University, Shebeen El-Kom 32511, Egypt

Corresponding author: Aly M. El-Semary (aelsemary@taibahu.edu.sa)

ABSTRACT Even though the ad-hoc on-demand distance vector (AODV) is a broadly used routing protocol for mobile ad-hoc networks (MANETs), it is vulnerable to a blackhole attack. Lu *et al.* developed a secure MANET routing protocol called SAODV to address the security weakness associated with the original AODV protocol and to remedy the blackhole attack. Specifically, the SAODV protocol can protect against blackhole attack performed by a malicious node during the routing process. However, it cannot resist the cooperative blackhole attack, in which two nodes are participating together to mount such attack. Therefore, this paper proposes a secure MANET routing protocol called BP-AODV to overcome the security breaches related to the SAODV protocol along with the original AODV protocol. In addition, the BP-AODV is able to protect against a cooperative blackhole attack launched during the routing process and guards against the blackhole attack that might take place during the forwarding process. The BP-AODV is developed by extending the functionality of the AODV protocol along with utilizing the chaotic map features. The experimental results assure that the BP-AODV protocol is more secure than the SAODV protocol and can effectively fight the blackhole attack achieved by a malicious node or cooperative malicious nodes during the routing process. The results also reveal that the BP-AODV can strongly guard against the blackhole attack that occurs during the forwarding process.

INDEX TERMS Cooperative blackhole attack, blackhole protected AODV, BP-AODV, malicious node, MANET.

I. INTRODUCTION

A mobile ad-hoc network (MANET) comprises a dynamic set of self-organizing mobile devices or nodes that directly communicate to each other without any fixed infrastructure. Thus, nodes in MANET perform the tasks of both hosts and routers to forward packets toward their destinations based on the employed routing protocol. Routing protocols utilized by MANET can be classified based on topology into: proactive, reactive, and hybrid protocols [1], [2].

In the proactive or table-driven routing protocols, each node has one or more routing tables to store entries for all available destination nodes. Nodes regularly advertise their routing tables information or directly after detecting a change in the network topology to maintain an up-to-date topology. These types of routing protocols experience a low delay but they suffer from adequate scalability. Examples of such protocols are the Destination Sequenced Distance Vector (DSDV) [3], its safe version called (SDSDV) [4], Optimized Link State Routing (OLSR) [5], and a lightweight Proactive Source Routing (PSR) [6] protocols. On the other hand, reactive or on-demand routing protocols create a route only when a source node has to send data to a destination node for which it has no entry in the routing table. These types of routing protocols have a good scalability but they experience a long delay. The Dynamic Source Routing (DSR) [7], its secure version called Secure Routing Protocol (SRP) [8], Ad-hoc On-demand Distance Vector (AODV) [9], and its non-cryptographic secure version (SAODV) [10] protocols are examples of reactive routing protocols. Finally, the hybrid routing protocols such as the ZRP (Zone Routing Protocol) [11] and its enhanced version IZRP (Independent Zone Routing Protocol) [12] divide the network topology into overlapping zones. They implement proactive protocols inside

The associate editor coordinating the review of this manuscript and approving it for publication was Xijun Wang.

each zone while they employ reactive protocols among the zones. These types of protocols take the advantages of the proactive and reactive protocols but they put an extra overhead on each node to maintain zone topology.

Due to the dynamic nature of MANETs and their lack of fixed infrastructure, they are generally vulnerable to several types of attacks. These attacks include sinkhole, DoS (Denial of Service), DDoS (Distributed DoS), and blackhole attacks [13]–[21]. Therefore, literature broadly covered such types of attacks. For example, Kalita *et al.* [13] surveyed different types of attacks associated with ad hoc networks and provided countermeasures for these attacks. Nguyen and Nguyen [14] introduced the impact of different types of attacks associated with MANETs based on a simulation study. Trivedi *et al.* [15] identified that ZRP is vulnerable to DDoS attack and then provided a new model for detecting the misbehaviour nodes. Hussain and Devaraj [16] analyzed the DSR protocol under the sinkhole attack and they concluded that the DSR is vulnerable to sinkhole attacks. Faghihniya *et al.* [17] ensured that AODV protocol is vulnerable to flooding attack which leads to DoS or DDoS. Panos *et al.* [18] presented a comprehensive analysis of the blackhole attack related to MANETs. In addition, they introduced the blackhole intensity as a new attack factor and evaluated its impact on the network performance. Khanna and Sachdeva [19] presented different aspects of blackhole attack together with the weaknesses of current literature. In addition, they introduced comprehensive classifications of the mitigation and detection schemes along with reviewing and comprising several published work associated with those classifications. Mejaele and Ochola [20] tested the DSR under the blackhole attacks and they found that the DSR is vulnerable to the blackhole attack. Thong and Buttyán [21] illustrated through examples that SRP (the secure version of DSR) is also vulnerable to blackhole attack.

Due to its good scalability along with low overhead, the AODV [9] protocol is one of the most widely employed reactive protocols. Thus, it becomes a main target for blackhole attack and cooperative blackhole attack. In the blackhole attack, a malicious node attracts network traffic due to an exploit in the route discovery process and then drops any data packets that are forwarded to it. This attack is generated in two steps: in the first step which takes place during a route discovery process, the malicious node will falsely advertise that it has the best up to date route to the destination. In the second step which takes place during the forwarding process, it will drop any data packets that are forwarded to it. The cooperative blackhole attack, on the other hand, is performed with two malicious nodes that cooperate together to lunch the attack.

To handle blackhole attack associated with the AODV, several variants of the AODV protocol have been proposed in the literature to mitigate such attack. Among these protocols is the non-cryptographic secure version of AODV called SAODV [10]. The SAODV protocol can protect only against blackhole attack performed by a malicious node. Unfortunately, as discussed in Section III-C, we found that the SAODV protocol is vulnerable to cooperative blackhole attack. Therefore, this paper addresses the vulnerability associated with the SAODV protocol that enables malicious nodes to lunch the cooperative blackhole attack. In addition, it proposes a blackhole protected AODV (BP- AODV) protocol that detects and protects against not only the blackhole attack but also the cooperative blackhole attack initiated during the routing process. The proposed protocol also provides protection against a malicious node that behaves normally during the routing process to avoid detection while it drops any data packets that might be received during the forwarding process. Further, it incorporates chaotic map into its design to guarantee that each distinct pair of nodes (i.e., source and destination) will have different secret parameters on each route request. The experimental results demonstrate the effectiveness of the proposed protocol compared with AODV, SAODV, and PCBHA [22].

The rest of this paper is organized as follows. Section II provides the related work while Section III presents the AODV and SAODV protocols along with their behavior under the blackhole attack. Section IV introduces the proposed routing protocol for MANETs. Section V shows the results of the conducted experiments. Finally, Section VI concludes the paper.

II. RELATED WORK

The AODV [9] is one of the mostly used reactive protocols. However, it is vulnerable to the blackhole attack which is a severe routing protocol attack as pointed in [2]. Specifically, Ochola *et al.* [23] analyzed the performance of the AODV protocol under blackhole attack scenarios. They pointed out that the protocol has very poor performance under blackhole attacks. Also, Jain and Choorasiya [24] along with Medadian *et al.* [25] experimentally showed that the performance of AODV protocol collapsed under blackhole attack. Accordingly, several solutions including [10], [22], [26]–[31] have been proposed to mitigate the blackhole attack associated with the AODV protocol.

Dokurer *et al.* [26] modified the AODV protocol to reduce the chance of a malicious or blackhole node to attach itself on a route. Specifically, the source node requesting the route ignores the first RREP packet or the first two RREP packets and then chooses next hop of any subsequent RREP packets because the blackhole node generally replies with RREP packet more quickly than any other nodes. This method is very useful to mitigate a blackhole attack performed with a single malicious node. Unfortunately, it has two shortcomings: 1) it is vulnerable to cooperative blackhole attack in which two malicious nodes are cooperating to launch the attack and 2) the protocol excludes the short path whenever there is no malicious nodes. Tamilselvan and Sankaranarayanan [27] changed the behavior of the source node in the AODV protocol to thwart the blackhole attack. Specifically, when a source node receives the first RREP packet, it does not immediately convey its data packet but it waits for a specific period of time to collect a set of RREP packets from its neighbor nodes. Next, the source node compares all RREP packets and chooses the neighbor node that has the same next hop as other alternative routes. Then, it starts the sending of its data packets.

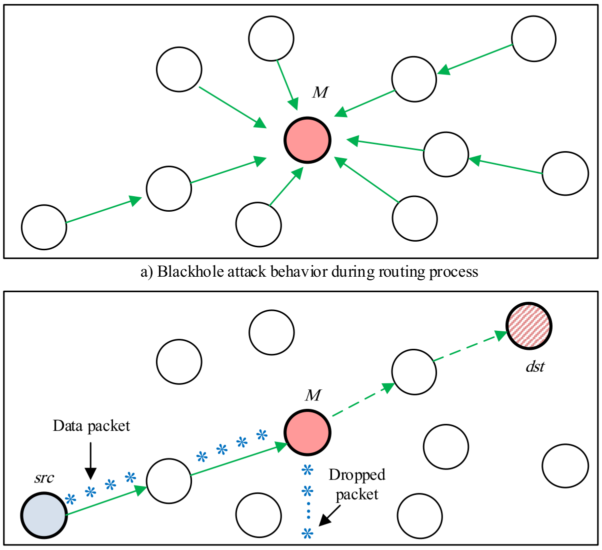
Tamilselvan and Sankaranarayanan [22] also developed a routing protocol based on the AODV to mitigate the cooperative blackhole attack. Their proposed protocol is called Prevention of a Co-operative Black Hole Attack (PCBHA). Initially, the PCBHA gives each node a default fidelity or trust level. When a source node broadcasts a RREQ message to establish a route, it waits to receive RREP messages from its neighboring nodes and then picks the neighboring node with a higher fidelity level that exceeds a predefined threshold value for forwarding the data packets to the destination node. When the destination node receives a data packet, it returns an ACK packet to the source node. If the source node receives an ACK packet, it will increment the fidelity level of the neighboring node from which it received the ACK message. Otherwise, it decrements the fidelity level and considers a possible blackhole node on this route. The PCBHA protocol builds its security based on receiving the ACK packets. If any malicious node replies with a forged ACK packet when it receives a data packet, the fidelity level of the route passing through the malicious node will be increased. This will lure the source node to establish a route through the malicious node which in turn can break the security of the PCBHA protocol and lunch the blackhole attack.

Choudhury *et al.* [28] introduced a Receive Reply method that utilizes a pre-RREP message that enables the source node to analyze the destination sequence number associated with the RREP packet and hence distinguishes fair nodes from malicious ones. Chavan *et al.* [29] provided a modified version of the AODV protocol to mitigate the blackhole attack. Their protocol expands the original AODV protocol using VERIFY and CHECKVRF messages initiated by the originator or source node to a destination node for verification purpose. When receiving a CHECKVRF packet, the destination responds with FINALREPLY packet to assure the authenticity of the path. Deshmukh *et al.* [30] developed a model that is based only on setting a validity bit in the RREP message. The model assumed that the malicious node has no knowledge about the validity bit that was sent in the RREP message. After receiving the RREP message, the source node checks the validity bit. If the validity bit is set to one, the source utilizes that path and starts the sending of its data packets. Otherwise, it considers that the route passes through a malicious node and hence ignores the RREP message. Yasin and Zant [31] discussed the shortcoming of this model. They argued that the model assumption is unrealistic because the malicious node employs the same protocol and can analyze it to recognize the validity bit before performing the attack. In addition, the model cannot specifically identify which node tries to perform the blackhole attack.

Lu *et al.* [10] developed a secure version of the AODV routing protocol called SAODV that withstands the blackhole attack associated with the original protocol. They extend the route discovery process of the AODV protocol with a verification process to directly verify a destination node. The verification process is achieved through exchanging random numbers between a source and a corresponding destination. As discussed in Section III-C, the protocol can protect only against the blackhole attack generated by a malicious node. Unfortunately, it is vulnerable to cooperative blackhole attack in which two malicious nodes are cooperated to initiate such attack.

III. AODV AND SAODV UNDER BLACKHOLE

This section briefly provides a background material about the blackhole attack in MANETs in Section III-A. In addition, it discusses the operations of the AODV [9] and SAODV [10] protocols under the blackhole attack in Sections III-B and III-C, respectively.



a) Blackhole attack behavior during routing process

b) Blackhole attack behavior during forwarding process

FIGURE 1. Blackhole attack behavior in MANETs.

A. MANETS BLACKHOLE ATTACK

The blackhole attack [13], [19], [32], [33] is a severe routing protocol attack that takes place during the routing process. In this attack, a malicious node M entices network nodes and advertises that it has the best path to a network destination during the routing process as visualized in Fig. 1a. As a result, a route including M on its path is created between a source node *src* and its associated destination node *dst*. When the *src* sends data packets to the *dst*, the packets will pass through M which in turn eavesdrops and then drops these packets as depicted in Fig. 1b. Also, two malicious nodes can cooperate together to perform the blackhole attack as discussed in [19], [34]. Furthermore, the blackhole attack can occur when a malicious node behaves normally (i.e., like a non-malicious node) during the routing process to avoid detection while it behaves abnormally when it is attached to a constructed route (i.e., it simply drops data packets during the forwarding process).

B. AODV PROTOCOL

The Ad-hoc On-demand Distance Vector (AODV) [9] is a reactive routing protocol for MANETs with two phases: route discovery and route maintenance. The AODV protocol utilizes four types of messages: Route Request (RREQ), Route Reply (RREP), Route Error (RERR), and hello (HELLO) to discover and maintain routes. The route discovery process is initiated only when a source node wants to send data to a destination node for which there is no entry in the routing table. To illustrate the discovery process, a network with sixteen nodes as shown in Fig. 2 is constructed. Suppose that node 2 is a source node denoted by *src* and node 15 is a destination node designated by *dst*. In addition, node 12 has a route to the *dst* through node 16 while node 14 has a direct route to the *dst*. Note that the text labels with gray color in figures refers to an operation executed in a previous round. For example, the label "RREQ during request round" in Fig. 2b refer to the route request performed in Fig. 2a.

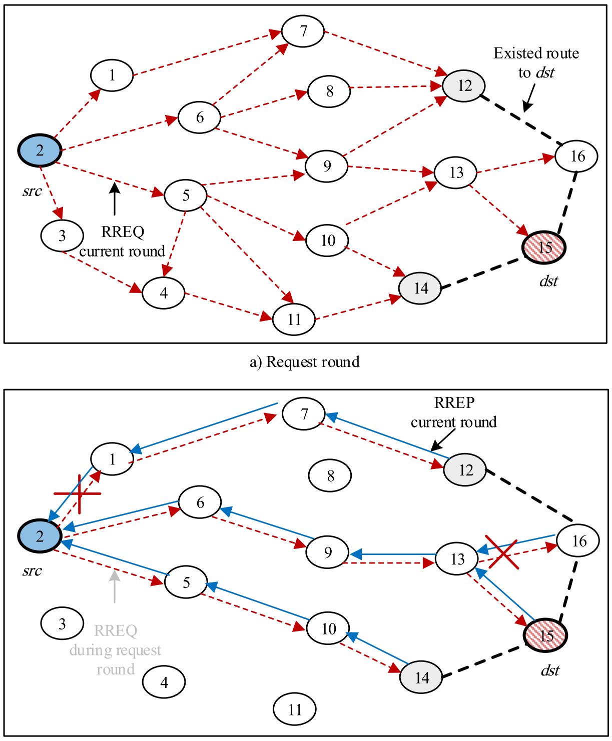


FIGURE 2. Route discovery process in AODV protocol.

During the discovery process shown in Fig. 2, three main steps are performed. Firstly, the *src* broadcasts a route request RREQ message as indicated in Fig. 2a to its next hop neighbors (nodes 1, 3, 5, and 6, in this case). Secondly, when an intermediate node that has no route to the *dst* (e.g., node 1, 3, 5, or 6) receives the RREQ message, it establishes a reverse route with the *src*. In addition, it broadcasts the RREQ message to its next hop neighbors after updating the received RREQ message (e.g., decrements the time-to-live field and increments the hop count field). This process is repeated until the RREQ message reaches either the *dst* or an intermediate node that has a route to the *dst*. When the *dst* or such intermediate node (e.g, node 12 or 14) receives the RREQ message, it also creates a reverse route to the *src* and then unicasts a RREP message to the *src* through the reverse route as indicated in Fig. 2b. Thirdly, when the *src* receives the RREP message, it chooses the route with the shortest number of hops and then starts the sending of data packets to the intended *dst*.

Even though AODV is a good and very popular routing protocol for MANETs, it is vulnerable to the blackhole attack [35]. In one scenario for the blackhole attack, suppose that node 10 is a malicious node denoted by *M* and the *src* broadcasts a route request RREQ message to the *dst* through intermediate nodes. When *M* receives the RREQ message, it lures the *src* by quickly replying with a RREP message indicating that it has the shortest route to the *dst*. As a result, the *src* receives the RREP message from *M* before any other reply messages. This forces the *src* to establish a virtual route to the *dst* through *M* and excludes the other routes as shown in Fig. 3 where each excluded route is indicated by a cross. In addition, when the *src* sends a data packet to the *dst*, the packet will pass through *M* which simply eavesdrops and then drops the packet to complete the blackhole attack. In other scenario for blackhole attack, *M* can behave normally during the routing process to avoid detection but if it is normally attached to a route, the malicious node will drop any data packet sent through that route.

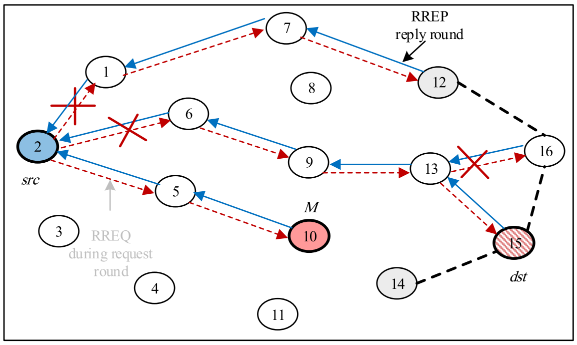
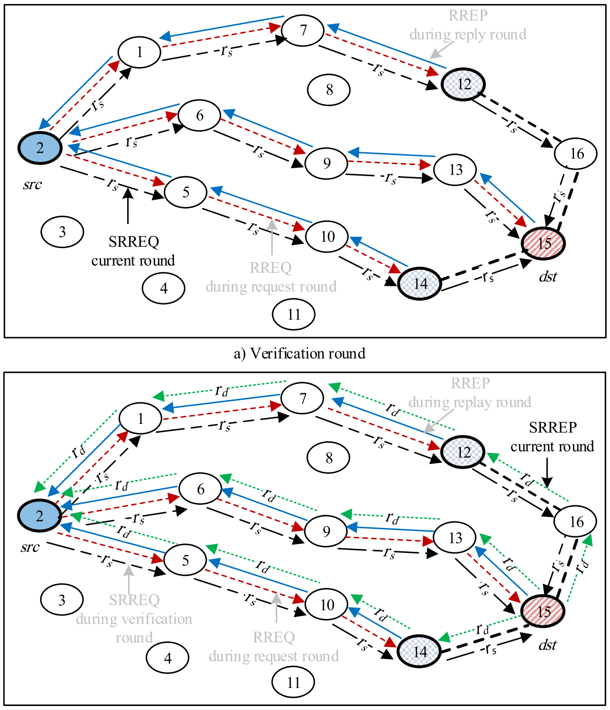


FIGURE 3. A blackhole attack scenario on AODV protocol.

C. SAODV PROTOCOL

The SAODV [10] is a secure routing protocol for MANET that is designed to protect against blackhole attack. Like the AODV protocol, the SAODV protocol has a route discovery phase and maintenance phase. The main difference between them is that the route discovery phase of the SAODV extends the functionality of the route discovery phase of the AODV with a verification process. The verification process enables the source node to directly verify the destination through exchanging random numbers between them. It is initiated when the source node receives a RREP message in the reply round of the route discovery phase of the AODV protocol. To illustrate the verification process, the same network field in Fig. 3 with the same assumptions is used.



a) Verification round

b) Verification confirm round

FIGURE 4. **Verification process in SAODV protocol.**

After the *src* receives a RREP in Fig. 2b, it stores the RREP in its routing table and directly sends a verification message SRREQ to the *dst* via the reverse direction of the route through which the RREP is received. The message SRREQ simply includes a random number r_s generated by the *src* as shown in Fig. 4a where the arrows started at the *src* node are labeled with r_s. When the *dst* receives at least two SRREQ messages on different routes, it stores the messages in its routing table and compares their content whether they have a same value of r_s. Based on the comparison results, the *dst* performs the following steps. Firstly, if at least two SRREQs have the same value of r_s, the *dst* sends a verification confirm message SRREP via the reverse direction of the route through which the SRREQ is received. The SRREP simply includes a random number r_d generated by the *dst* as indicated by an arrow labeled with r_d and started from the *dst* in Fig. 4b. For example, the *dst* received SRREQ messages from the routes through nodes 13, 14, and 16 that have the same value r_s as depicted in Fig. 4a. Therefore, the *dst* sends a SRREP with r_d to each of the nodes 13, 14, and 16 as indicated by arrows labeled with r_d in Fig. 4b. Secondly, if the SRREQ messages have different r_s values, the *dst* needs to wait until it receives at least two SRREQ messages that have the same r_s value and then performs the first step. On the other hand, when the *src* receives two SRREP messages that have the same r_d value from different routes, it chooses the route with the shortest number of hops to the *dst*. After selecting the verified shortest route, the *src* starts its transmission to the *dst*.

From security perspective, the SAODV protocol is designed to protect MANETs against the blackhole attack. That is, the protocol can handle the blackhole attack of a single malicious node. Unfortunately, the SAODV is vulnerable to cooperative blackhole attack which can be performed by two malicious nodes. Specifically, if two malicious nodes are cooperated by sending the same confirmation random number during the verification confirm round, they will entice the source node to select one of the routes on which the malicious nodes are attached. One scenario to lunch blackhole attack on SAODV is illustrated on the same network in Fig. 5 with the same previous assumptions. In addition, suppose that nodes 8 and 10 are two malicious nodes denoted by M_1 and M_2, respectively.

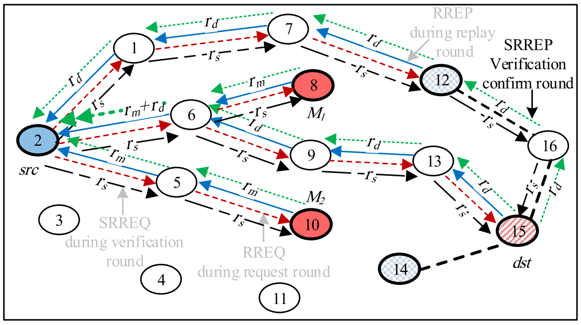


FIGURE 5. **Blackhole attack scenario on SAODV protocol.**

In general, when the *src* requests a route to the *dst*, it broadcasts a RREQ message through intermediate nodes during the request round as indicated in Fig. 5. According to the protocol steps, each of M_1, M_2, node 12, and *dst* will respond with a RREP message during the reply round. When the *src* receives a RREP message, it directly responds via the RREP reverse route with a SRREQ during the verification round. The SRREQ contains a random number r_s generated by the *src* as indicated in Fig. 5 by an arrow labeled with r_s. M_1 and M_2 will cooperate to respond with a SRREP that has the same random number r_m as indicated by an arrow labeled with r_m started from M_1 and M_2. In addition, the *dst* responds with a SRREP that has r_d through nodes 13 and 16 as depicted by an arrow labeled with r_d initiated from the *dst* in Fig. 5. Note that there is an arrow with two heads from node 6 to the *src*. This arrow means that node 6 conveys two SRREP messages with different random numbers, one received from M_1 and the other from node 9. Since each of the two malicious nodes has less number of hops to the *src* than node 12 and the *dst*, the *src* will receive the two SRREP sent by the malicious nodes before the other SRREP's. As a result, the *src* will establish a route with either M_1 or M_2. Also, the SAODV protocol does not consider the blackhole attack that is achieved by a malicious node that behaves normally during the routing process while it behaves abnormally during the forwarding (i.e., such node will drop the received packets passing through which to the destination).

IV. PROPOSED BP-AODV ROUTING PROTOCOL

The Blackhole Protected Ad-hoc On-demand Distance Vector (BP-AODV) routing protocol is developed to protect

MANETs against blackhole attack in general while it specially overcomes the blackhole attack associated with the AODV and SAODV protocols. The main contribution of the proposed protocol is to address the cooperative blackhole attack associated with the SAODV protocol along with providing a routing protocol that is robust against not only the blackhole attack but also the cooperative blackhole attack during the routing process. The BP-AODV can detect malicious nodes that behave abnormally during the route discovery process. In addition, it guards against blackhole attack that might be performed by a malicious node that behaves normally (i.e., the malicious node follows the protocol steps) during the routing process but it behaves maliciously during the forwarding process.

The proposed protocol uses a challenge-response-confirm pattern to establish trusted routes. In general, a source node generates a challenge value and then conveys it to a destination node during a route request. When the destination receives the challenge, it computes the corresponding response as a function of the received challenge along with other secret values generated by the destination. The destination node propagates the response value to the source during the route reply while it keeps the secret values. Finally, the destination node confirms the route by conveying the secret values during the route confirm. The BP-AODV accomplishes its task by incorporating chaotic map into its design along with using the challenge-response-confirm pattern during routing process as discussed in Section IV-B. In addition, it calculates the degree of a node trust based on the number of forwarded data packets during the forwarding process as introduced in Section IV-C. The BP-AODV protocol assumes that the nodes within a transmission range of each other have a number of benign nodes greater than the number of malicious nodes. Before going to the protocol details, the BP-AODV utilizes five types of messages as presented in Section IV-A.

A. BP-AODV MESSAGE TYPES

The proposed BP-AODV protocol employs five types of messages: MRREQ (Modefied Route REQuest), MRREP (Modefied Route REPly), RERR (Route ERRor), HELLO (HELLO), and RCON (Route CONfirm). Actually, BP-AODV adapts RREQ and RREP while it uses the RERR and HELLO messages [36] of AODV protocol. In addition, it develops the RCON message as a new message type.

The MRREQ message extends the RREQ message with two fields referred to as c_s and t_s. The field c_s holds a challenge value generated by a source node while the field t_s keeps the time in millisecond at which the MRREQ message is created. Also, the MRREP message expands the RREP message with one field denoted by v to accommodate a response value v calculated by a destination node. Finally, the constructed RCON message has eight fields to enable the destination conveying the secret values during the route confirm (i.e., the values used to confirm the correctness of the response value propagated during the route reply). The name and description of each RCON message field are introduced in Table 1.

TABLE 1. The name and description of each RCON field.

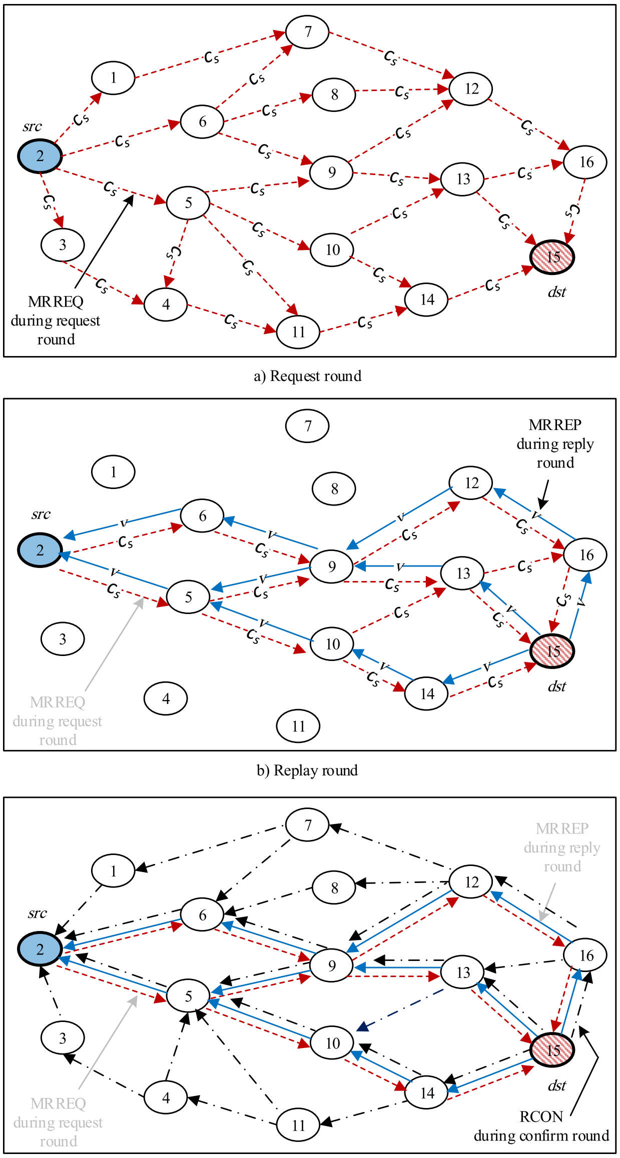
	Name	Description
1	Type	The message type
2	DIP	IP address of the node requested the route
3	SIP	IP address of the node generated the RCON message
4	η	Secret value used as the number of chaotic map iterations
5	r_1	The first secret random number
6	r_2	The second secret random number
7	hc	Number of hops from the dst to the node receiving RCON
8	t_c	Time stamp at which the RCON message is created

B. BP-AODV ROUTING PROCESS

In general, the proposed BP-AODV protocol protects MANET's against blackhole attack and cooperative blackhole attack and specially, it remedies the blackhole vulnerability of the SAODV and AODV protocols. In addition, the protocol detects malicious nodes that behave abnormally during the routing process. Like the AODV and SAODV, the BP-AODV has a route discovery phase and a route maintenance phase. The route discovery phase of the BP-AODV protocol extends the functionality of the route discovery phase of AODV protocol to implement the challenge-response-confirm pattern along with establishing up to three routes instead of one route by the AODV protocol. The route discovery phase of the BP-AODV protocol is achieved by completing three rounds or processes: *Request*, *Reply*, and *Confirm*.

To illustrate each process, a network of sixteen nodes shown in Fig. 6 is constructed. Also, suppose that node 2 is a source node denoted by *src* and node 15 is a destination node designated by *dst*. The BP-AODV protocol is a reactive protocol so the *src* requests a route only when it has no entry in its routing table for the *dst*.

The *Request* process indicated in the left-top part of Fig. 7 is initiated by the *src* to construct a route with the *dst*. During this round, reverse routes with the *src* are established and a challenge value c_s generated by the *src* is conveyed to the network nodes. Specifically, the *src* creates a MRREQ message together with putting parameter values (e.g., c_s, hop count *hc*, and broadcast id *bid*) into corresponding fields of the message. Then the *src* broadcasts the MRREQ message to its neighbor nodes as indicated by dashed arrows labeled with c_s to nodes 1, 3, 5, and 6 in Fig. 6. When an intermediate node receives the MRREQ message, it ignores the message if it is previously received from the same forwarding node or the number of previously received messages from different nodes is greater than three. Otherwise, the intermediate node performs four steps. Firstly, it increments the number of received MRREQ messages. Secondly, it builds a reverse route with the *src* through the forwarding node. Note that the BP-AODV allows up to three routes with the *src* via different forwarding nodes. Thirdly, it stores the challenge value c_s into its routing table. Fourthly, if the intermediate node is not the *dst* and the MRREQ is not previously broadcasted, the node increments the number of hop count *hc* included in the message and then



a) Request round

b) Replay round

c) Confirm round

FIGURE 6. BP-AODV operations during routing process.

broadcasts it to the neighbor nodes. This process is repeated by all intermediate nodes until the MRREQ message reaches the *dst* which in turn starts the *Reply* process.

On the other hand, the *Reply* process shown in the bottom part of Fig. 7 is started by the *dst* immediately after receiving the first MRREQ message. During this round, a response value *v* calculated by the *dst* is propagated to nodes on the route paths and reverse routes with the *dst* are established. Also, the protocol allows the receiving of only one MRREP message from the same node along with a maximum of three from different nodes. Specifically, when receiving a MRREQ message, the *dst* executes a maximum of four main steps. Firstly, if the same message is previously received, the *dst* skips this step and starts the execution from the second step. Otherwise, the *dst* calculates a response value *v* based on Equation 1.

$$v = \lfloor x(\eta) * 10^{14} \rfloor \tag{1}$$

where the symbol $\lfloor\ \rfloor$ is the Math floor function that rounds its argument while $x(\eta)$ is the Logistic chaotic map [37], [38]. The Logistic map is widely employed in several applications for securing multimedia contents. The BP-AODV incorporates the Logistic map in its design to inherit its brilliant features such as ergodicity, randomness, and sensitivity to initial conditions and control parameters. Accordingly, the parameters used in calculating the response value and confirmation of the route are not fixed and totally related to the *src* and *dst* nodes in each route request. In fact, the utilization of the chaotic map grants the proposed protocol more security. The chaotic value $x(\eta)$ [38] can be computed by Equation 2.

$$x(\eta) = \mu x(\eta - 1)[1 - x(\eta - 1)] \tag{2}$$

where η is a secret value randomly generated by the *dst* to represent the number of chaotic map iterations. $x(0) \in (0, 1)$ and $\mu \in (0, 4]$ are the initial value and the control parameter of the Logistic map, respectively. To assure the good chaotic behavior of the map, the parameter μ should be maintained in the range [3.5699456, 4] [38]. Accordingly, the values of $x(0)$ and μ can be obtained from the developed Equations 3 and 4, respectively.

$$x(0) = (\frac{c_s}{r_1})\ mod\ 1 \tag{3}$$

$$\mu = 3.5699456 + 0.43((\frac{c_s}{r_2})\ mod\ 1) \tag{4}$$

where c_s is a challenge value conveyed by the *src* while r_1 and r_2 are two secret values randomly generated by the *dst*. Note that Equations 3 and 4 are developed to generate and maintain the initial value $x(0)$ and the control parameter μ in their valid range [38] (i.e. $x \in (0, 1)$ and $\mu \in [3.5699456, 4]$). Thus, the generated chaotic values will carry the good features of the employed map. In addition, both equations relate the generated parameters with the challenge value c_s produced by the *src* node and with the secret values r_1 and r_2 created by the *dst* node. In other words, the generated values are not fixed and totally depend on the pair of nodes (i.e., *src* and *dst*). Accordingly, each distinct pair of nodes will have different parameters on each new route request which grant the protocol more security.

Secondly, the *dst* updates the timer *t* based on Equation 5.

$$t = \frac{\sum_{i=1}^{n}(t_{r_i} - t_s)}{2n} \tag{5}$$

where t_s is the time at which the MRREQ is generated by the *src* while t_{r_i} is the time at which the i^{th} MRREQ message received by the *dst*. *n* is the number of received MRREQ messages with the same *bid*. The maximum number of messages allowed to be received is three, each one from a different forwarding node (i.e., $1 \leq n \leq 3$).

Thirdly, the *dst* creates a corresponding MRREP message and sets up the required message parameters including the

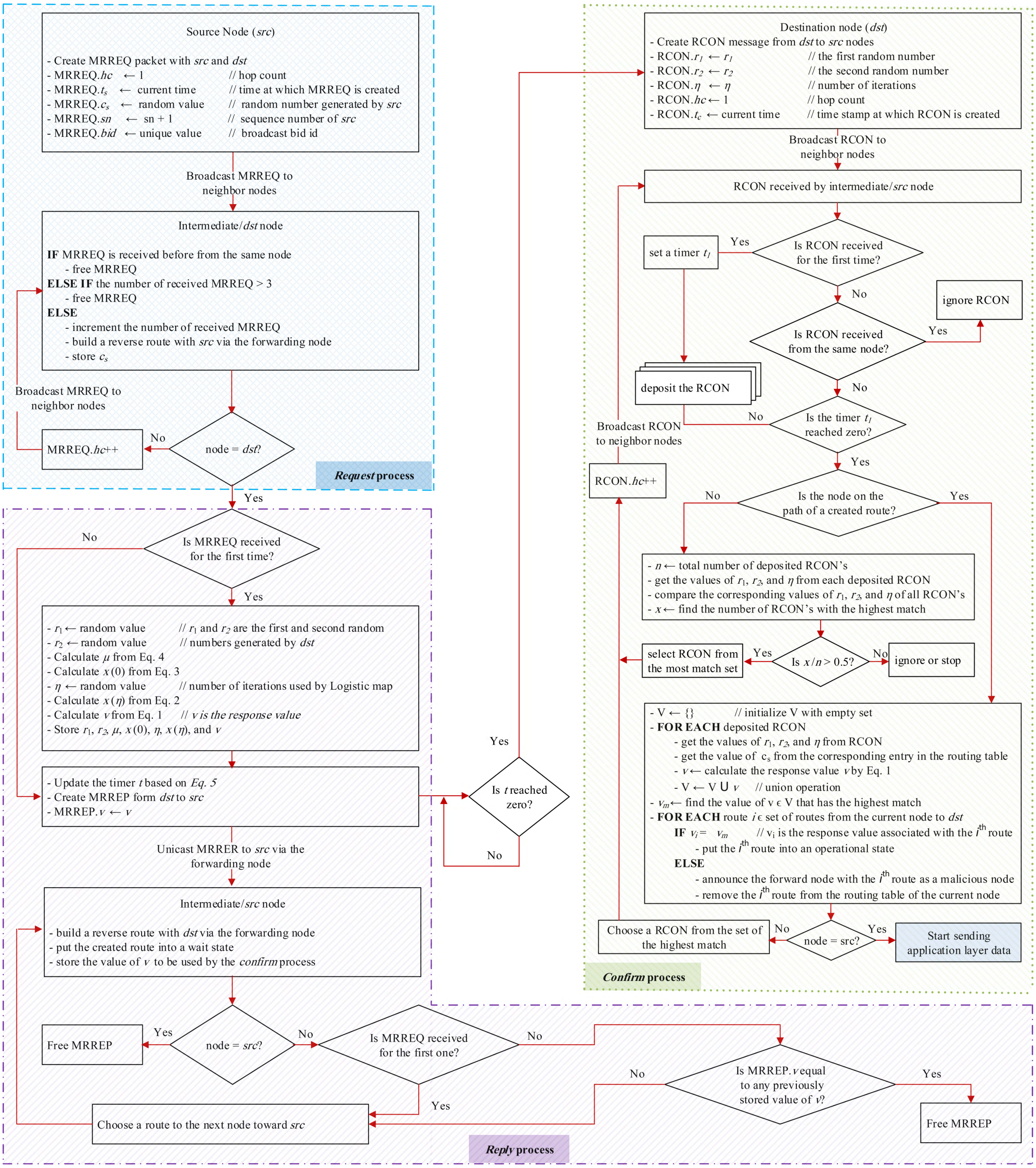

Source Node (src)
- Create MRREQ packet with src and dst
- MRREQ.hc ← 1 // hop count
- MRREQ.t_s ← current time // time at which MRREQ is created
- MRREQ.c_s ← random value // random number generated by src
- MRREQ.sn ← sn + 1 // sequence number of src
- MRREQ.bid ← unique value // broadcast bid id
Broadcast MRREQ to neighbor nodes
Intermediate/dst node
IF MRREQ is received before from the same node
- free MRREQ
ELSE IF the number of received MRREQ > 3
- free MRREQ
ELSE
- increment the number of received MRREQ
- build a reverse route with src via the forwarding node
- store c_s
Broadcast MRREQ to neighbor nodes
MRREQ.hc++
No
node = dst?
Request process
Yes
No
Is MRREQ received for the first time?
Yes
- r1 ← random value // r1 and r2 are the first and second random
- r2 ← random value // numbers generated by dst
- Calculate μ from Eq. 4
- Calculate x(0) from Eq. 3
- η ← random value // number of iterations used by Logistic map
- Calculate x(η) from Eq. 2
- Calculate v from Eq. 1 // v is the response value
- Store r1, r2, μ, x(0), η, x(η), and v
- Update the timer t based on Eq. 5
- Create MRREP form dst to src
- MRREP.v ← v
Yes
Is t reached zero?
No
Unicast MRRER to src via the forwarding node
Intermediate/src node
- build a reverse route with dst via the forwarding node
- put the created route into a wait state
- store the value of v to be used by the confirm process
Free MRREP
Yes
node = src?
No
Is MRREQ received for the first one?
No
Yes
Choose a route to the next node toward src
No
Is MRREP.v equal to any previously stored value of v?
Yes
Free MRREP
Reply process
Destination node (dst)
- Create RCON message from dst to src nodes
- RCON.r1 ← r1 // the first random number
- RCON.r2 ← r2 // the second random number
- RCON.η ← η // number of iterations
- RCON.hc ← 1 // hop count
- RCON.t_c ← current time // time stamp at which RCON is created
Broadcast RCON to neighbor nodes
RCON received by intermediate/src node
set a timer t1
Yes
Is RCON received for the first time?
No
ignore RCON
Yes
Is RCON received from the same node?
deposit the RCON
No
No
Is the timer t1 reached zero?
Broadcast RCON to neighbor nodes
Yes
RCON.hc++
No
Is the node on the path of a created route?
Yes
- n ← total number of deposited RCON's
- get the values of r1, r2, and η from each deposited RCON
- compare the corresponding values of r1, r2, and η of all RCON's
- x ← find the number of RCON's with the highest match
select RCON from the most match set
Yes
Is x/n > 0.5?
No
ignore or stop
- V ← {} // initialize V with empty set
- FOR EACH deposited RCON
- get the values of r1, r2, and η from RCON
- get the value of c_s from the corresponding entry in the routing table
- v ← calculate the response value v by Eq. 1
- V ← V ∪ v // union operation
- v_m ← find the value of v ϵ V that has the highest match
- FOR EACH route i ϵ set of routes from the current node to dst
IF v_i = v_m // v_i is the response value associated with the i^th route
- put the i^th route into an operational state
ELSE
- announce the forward node with the i^th route as a malicious node
- remove the i^th route from the routing table of the current node
Choose a RCON from the set of the highest match
No
node = src?
Yes
Start sending application layer data
Confirm process

FIGURE 7. The flowchart of the routing process of the BP-AODV protocol.

value of v along with keeping the secret values η, r_1, and r_2. Fourthly, the *dst* unicasts the MRREP message to the *src* through the forwarding node (i.e., the node from which the *dst* received the MRREQ message). For example, the *dst* unicasts a MRREP message to the *src* through each of the forwarding nodes 13, 14, and 16 as indicated by the arrows labeled with v in Fig. 6b.

When an intermediate node receives a MRREP message, it carries out four main steps. Firstly, it establishes a reverse route to the *dst* through the forwarding node. The node is allowed to create up to three routes with the *dst* (e.g., node 9 in Fig 6b created two reverse routes to the *dst* through nodes 12 and 13). Secondly, the node puts the established route into a waiting state (i.e., the route cannot be used to forward data packets until it is turned into an operational state). A route is turned into the operational state by the *Confirm* process only if it assures that the route has no malicious node on its path. Thirdly, the node stores the value of v into its routing table with the associated route. Fourthly, if the intermediate node is the *src*, it ignores the message. Otherwise, the node unicasts each of the received MRREP messages having the same value of v to the *src* through different routes (e.g., node 9 receives two MRREP messages with the same value of v so it unicasts one message via node 12 and the other message through node 13). On the other hand, if the node receives MRREP messages having different values of $v's$, it unicasts these messages to the *src* through the same node.

Finally, the *Confirm* process depicted in the right-top part of Fig. 7 is triggered by the *dst* when the value of its timer t reaches zero. During the confirm round, the BP-AODV achieves three tasks: 1) the BP-AODV reveals the secret values of η, r_1, and r_2 stored at the *dst* to the network nodes, 2) it detects malicious nodes tried to perform the blackhole attack, and 3) it removes the routes that have malicious node on their paths while assuring the other routes.

Specifically, when the value of its timer t reaches zero, the *dst* creates a corresponding RCON message that includes the secret values of η, r_1, and r_2. Next, the *dst* broadcasts the RCON message to its neighbor nodes as denoted by three dashed-dotted arrows in Fig. 6c started from *dst* to nodes 13, 14, and 16. When an intermediate node that is not on a route path (nodes 1, 3, 4, 7, 8 or 11 in Fig. 6c) receives a set of RCON messages from different nodes in a specified period of time, it compares the corresponding values of η, r_1, and r_2. If the majority of RCON messages have the same corresponding values, the node broadcasts only one of these majority messages. Otherwise, if the number of the RCON messages having the same values is neutral, the node ignores the messages. In contrast, when an intermediate node that is on a route path (nodes 5, 6, 9, 10, 12, 13, 14, or 16) receives a set of RCON messages from different nodes in a specified period of time, it performs five main steps. Firstly, the node calculates the value of v from Equation 1 for each received RCON message based on its included values of η, r_1, and r_2 along with the value of c_s in the node routing table. Secondly, the node compares all the values of v to get the most match value denoted by v_m. Thirdly, the node compares the value of v_m with each value v_i associated with the i^{th} route in its routing table. Fourthly, if the values of v_m and v_i are equal, the node turns the i^{th} route into an operational state. Otherwise, the node announces that the forwarding node associated with the i^{th} route is a malicious node and then removes the i^{th} route from its routing table. Lastly, if the node is not the *src*, it broadcasts the one most match with v_m to its neighbor nodes. Otherwise, the node (i.e., *src*) starts to forward data packets to the *dst* based on the developed forwarding process discussed in the next section.

C. BP-AODV FORWARDING PROCESS

The BP-AODV introduces a new forwarding process as a second layer of defense to protect MANET's against the blackhole attack achieved by a malicious node that behaves normally (i.e., the malicious node follows the protocol steps) during the routing process but it behaves abnormally during the forwarding process. The forwarding process starts immediately by the *src* after approving its routes with the *dst* during the routing process. During the forwarding process, when the *src* or any forwarding node on route paths to the *dst* wants to forward a data packet to the *dst*, it selects next hop n_i toward the *dst* based on the developed Equation 6.

$$n_i = \begin{cases} f(x,r) & \text{if } y_1 \text{ to } y_r \text{ are equal} \\ 1 & 0 < x \le \dfrac{y_1}{\sum_{j=1}^{r} y_j} \\ 2 & \dfrac{y_1}{\sum_{i=1}^{r} y_i} < x \le \dfrac{y_1 + y_2}{\sum_{j=1}^{r} y_j} \\ 3 & x > \dfrac{y_1 + y_2}{\sum_{j=1}^{r} y_j} \end{cases} \tag{6}$$

where $x \in (0, 1)$ is a random number generated by the forwarding node. r is an integer representing the number of routes toward the *dst* at the forwarding node and its value is 1, 2, or 3. For example, the *src* has two routes (i.e., $r = 2$) to the *dst* through the next hop nodes 5 and 6 as shown in Fig. 6c. y_j is the number of received data packets from the next hop n_j associated with the j^{th} route at the forwarding node. The function $f()$ is evaluated by Equation 7.

$$f(x, r) = 1 + \lfloor x * 10^3 \rfloor \bmod r \tag{7}$$

The function $f()$ is used to randomly select a next hop when all next hops have the same degree of trust at the forwarding node (i.e., all values of y_j are equal for $1 \le j \le r$). The value of y_j associated with the next hop n_j provides a degree of trust for the j^{th} route. To illustrate Equation 6, suppose that node 5 wants to forward a data packet to *dst*. Also node 5 previously received 70 and 30 data packets from nodes 9 and 10 (i.e., $y_1 = 70$ is associated with the 1^{st} route via node 9 while $y_2 = 30$ is related to the 2^{nd} route through node 10), respectively. Thus, the degree of trust of the 1^{st} and 2^{nd} routes are 0.7 (i.e., 70/ (70 + 30)) and 0.3 (i.e., 30/ (70 + 30)), respectively. Next, the value of $x \in (0, 1)$ determines which route is going to be selected. In this example, if the value of $x \le 0.7$, the 1^{st} route is selected. Otherwise, the 2^{nd} route is chosen. This means that the higher the trust degree of a route, the higher the chance of the route to be selected. Note that the protocol does not directly select the route of the highest degree of trust for two reasons: 1) the forwarding node increases its trust at the next hop and 2) it makes load balance between different routes.

Based on Equation 6, the protocol gives a very rare chance to forward data packets through a malicious node that performs blackhole attack during the forwarding process. This comes from the fact that a malicious node that performs the blackhole attack does not forward data packets (i.e., the number of received data packets from the malicious node is zero). Therefore, the degree of trust of the route through the malicious node is zero.

D. BP-AODV BLACKHOLE PROTECTION

The blackhole attack can be performed in several scenarios. Firstly, a malicious node puts itself on a route path during the routing process as depicted in Fig. 3. Secondly, two malicious nodes can cooperate to attach one of them on the route path during the routing process as presented in Fig. 5. Thirdly, a malicious node follows the protocol steps during the routing process but it behaves maliciously (e.g., it drops any received data packets) during the forwarding process. Since the second scenario somehow covers the first scenario, only the protection against the second and third scenarios are discussed.

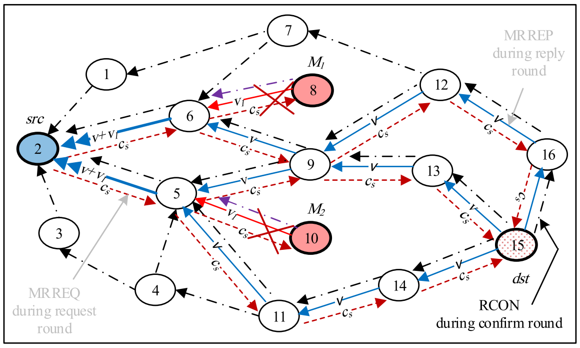


FIGURE 8. Detecting and removing malicious nodes.

The BP-AODV protects against cooperative balckhole attack. To illustrate the protection for the second scenario, the same network in Fig. 6 is considered for which nodes 8 and 10 are utilized as malicious nodes M_1 and M_2, respectively. During the request and reply rounds, the *src* and *dst* build a set of routes between them and exchange the challenge and response values as shown in Fig. 8. The *dst* conveys a response value denoted by v while the two malicious nodes cooperate with each other to send the same response value denoted by v_1. Note that each of nodes 5 and 6 has an arrow with two heads to *src*. This arrow means that a node conveyed two MRREP message, each with different response value. For example node 6 received messages contain the same value of v from nodes 7 and 9 while it received a message with the value v_1 from M_1. According to the protocol steps, all benign nodes put their established routes into a wait state so the *src* cannot enticed by the malicious nodes at this moment even if the malicious nodes start their confirmation round. During the confirm round, the *dst* assures routes by revealing the secret values that are used in calculating the conveyed v. For example, node 6 receives two RCON messages from nodes 7 and 9 that have the same confirm values while it receives a different RCON from M_1. According to the protocol steps, node 6 concludes that node 8 is a malicious node. Therefore, it removes node 8 from its routing table while turning the route via node 9 into an operational state. Then, it broadcasts either the RCON message received from node 7 or 9 to the *src*. In this case, the *src* receives four benign RCON messages that lead to assure routes through nodes 5 and 6. Note that node 2 (*src* in this case) will also detect that each of nodes 5 and 6 is recovered from malicious nodes.

The BP-AODV protocol also thwarts blackhole attack in the third scenario. Due to the fact that a malicious node performing the blackhole attack does not forward any data packets to other nodes, it does not build a trust with the nodes. The developed Equation 6 is designed with this fact to avoid the selection of the malicious node as a next hop during the forwarding process. This second layer of protection introduced in Section IV-C increases the robustness of the BP-AODV protocol against the blackhole attack. These three scenarios are experimentally tested with the BP-AODV protocol and the results assure the robustness of the proposed protocol against MANET's blackhole attack.

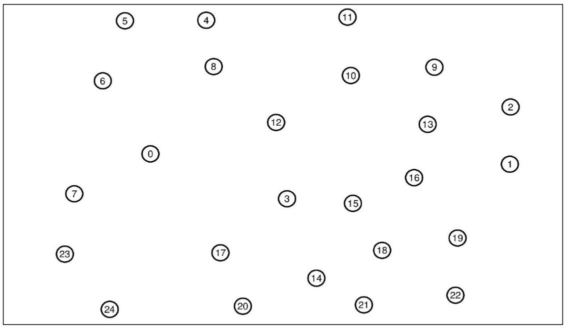


FIGURE 9. Network field used in experiments.

V. EXPERIMENTAL RESULTS

This section presents a set of conducted experiments that compares the performance of the proposed BP-AODV protocol against the SAODV [10], AODV [9], and PCBHA [22] protocols under both attack-free and blackhole attack conditions. The performance is measured in terms of throughput, packet delivery ratio (PDR), and end-to-end delay. The throughput is the amount of data received in a given time period and it is usually measured in bits per second. The PDR is the ratio of the number of packets received to the number of packets sent by the destination and source nodes, respectively. The end-to-end delay is the time that a packet takes from its source until reaching its destination [39], [40]. The experiments are implemented and performed using the popular network simulator 2 (NS2) [41] and they are carried on several network fields of different sizes. It is found that the obtained results from these different network fields exhibit similar performance behavior. Therefore, a simple network field is provided to show the performance behavior of the underling protocols for attack-free and blackhole attack conditions. The network field of the simulator has 25 nodes distributed over an area of size $1000m \times 500m$ and its initial node positions are depicted in Fig. 9.

As discussed in Section III, the blackhole or cooperative blackhole attack can be achieved in three main scenarios. Firstly, the blackhole attack can be lunched by a malicious node that behaves maliciously to attach itself on a route path during the routing process and drops its received data packets during the forwarding process. Secondly, the cooperative blackhole attack can be generated by using two malicious nodes that cooperate with each other to attach one of them on the route path during the routing process while dropping data packets during the forwarding process. Thirdly, the blackhole attack can be accomplished by a malicious node that behaves

TABLE 2. Simulation configuration parameters of each scenario.

Parameters \ Scenarios		Scenarios 1 and 2			Scenario 3		
		con1	*con2*	*con3*	*con1*	*con2*	*con3*
Simulation	*Src*	4	10	6	16	18	4
	Dst	0	2	22	10	11	21
	Start time (s)	1	20	50	1	20	40
	End time (s)	20	65	200	60	65	200
	Rate (Mbs)	0.1	0.1	0.01	0.01	0.01	0.01
	Packet size (B)	1000	1000	1000	1000	1000	1000
	Traffic type	CBR			CBR		
	Duration (s)	200			200		
	speed (m/s)	25			20		
	Number of nodes	25			25		
	Size of Network field	$1000m \times 500m$					
Physical layer	*S. propagation*	Two-ray ground					
	Antenna model	Omni antenna					
Mac layer	Mac protocol	802.11					
	Link bandwidth	1 MB (defualt)					
Queue	Type	DropTail/PriQueue					
	Size	50					
NS2 version		2.35					
Processor		Intel processor 3GH					
Operating System		Ubuntu 16.04 LTS					

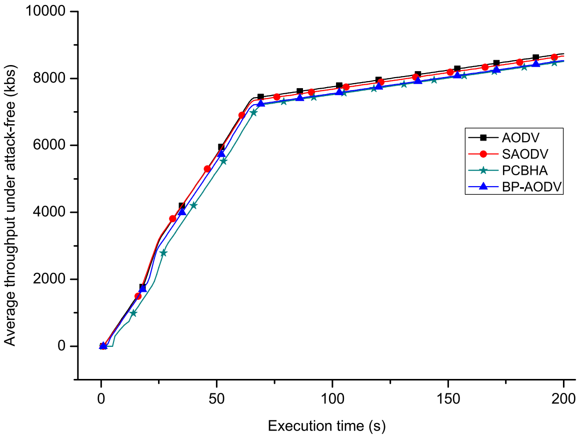


FIGURE 10. Attack-free average throughput in the 1^{th} scenario.

normally during the routing process to avoid detection while it behaves maliciously during the forwarding process by simply dropping its received data packets. Therefore, the conducted experiments are divided into three categories, one for each scenario. The simulation configuration parameters of each scenario are described in Table 2. For example, each of the first and second scenarios has three connections denoted by $con1$, $con2$, and $con3$ that connect the source nodes 4, 10, and 6 to the destination nodes 0, 2, and 22, respectively. Also, the duration lasts for 200 seconds in which $con1$, $con2$, and $con3$ begin at 1, 20, and 50 while they end at 20, 65, and 200 from the simulation start time, respectively. In addition, the other simulation parameters along with the parameters related to the physical layer, MAC layer, and queue are shown in the table. After setting up the required configurations, the experiments of each scenario are executed to measure the performance of the BP-AODV protocol against the SAODV, PCBHA, and AODV protocols.

During the first scenario, two experiments are performed to evaluate the performance of the four protocols in terms of the average throughput, end-to-end delay, and PDR. The experiments use a step size of one second to calculate the average. The first experiment measures the performance of the four protocols under attack-free condition. Since the four protocols perform well under the attack-free condition, only their average throughput results are presented in Fig. 10.

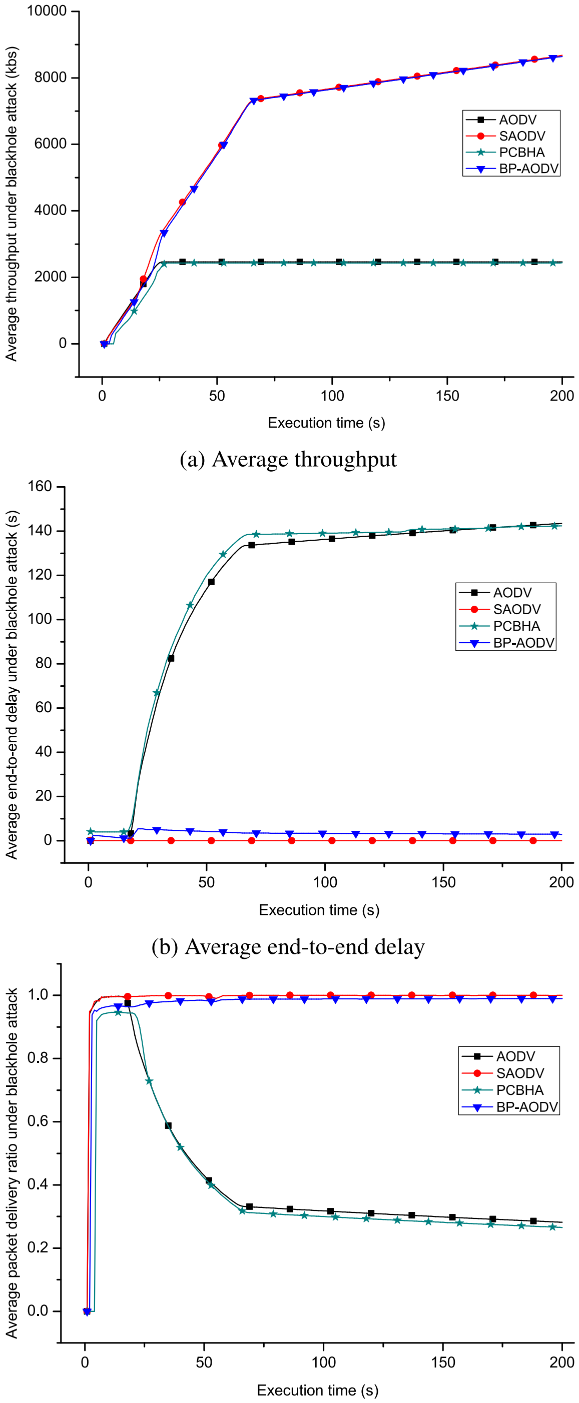
The results reveal that the four protocols produce almost the same average throughput under attack-free condition. On the other hand, the second experiment computes the performance under blackhole attack generated by a malicious node during the routing process. The experiment uses node 3 in Fig. 9 as a malicious node and its results are shown in Fig. 11.

The results of the average throughput in Fig. 11a, end-to-end delay in Fig. 11b, and PDR in Fig. 11c show that the BP-AODV and SAODV protocols are protected under the blackhole attack performed with one malicious node during the routing process. Both protocols can detect the malicious node and remove it from the route paths during the routing process. In the PCBHA and AODV protocols, on the other hand, the malicious node is able to lunch the blackhole attack because the AODV protocol is not considered the security in its implementation while the PCBHA considers the security in its implementation but it is based on sending an ACK packet which is forged by the malicious node during the attack. Therefore, in each of the PCBHA and AODV protocol, the malicious node is able to put itself on the route paths of the connections $con2$ and $con3$ during the routing process. When the malicious node receives data packets on $con2$ or $con3$ during the forwarding process, it drops them. In addition, the malicious node in the PCBHA protocol forges an ACK packet and sends it to the source node to increase its fidelity or trust level. Dropping the data packets by the malicious node in the PCBHA and AODV protocols results in reducing their performance as shown in Fig. 11.

During the second scenario which is concerned with cooperative blackhole attack (i.e., two malicious nodes are cooperated to perform that attack), two experiments employing nodes 3 and 16 in Fig. 9 as malicious nodes are carried out. The first experiment visualizes the average end-to-end delay under both the blackhole attack-free and the cooperative blackhole attack conditions of the four protocols to just give an indication about the overhead of the BP-AODV protocol compared with the other three protocols. The experimental results in Fig. 12 reveal that the average end-to-end delay of the four protocols under attack-free is almost close to each other. In contrast, the results under cooperative blackhole attack show that the BP-AODV protocol experiences very low average end-to-end delay compared with the SAODV, PCBHA, and AODV protocols.

The second experiment of the second scenario which has the results in Fig. 13 measures the performance of the four protocols under the cooperative blackhole attack.

The results of the average throughput in Fig. 13a, end-to-end delay in Fig. 13b, and PDR in Fig. 13c indicate that the BP-AODV protocol mitigates the effect of cooperative blackhole attack while the AODV and SAODV are



(a) Average throughput

(b) Average end-to-end delay

(c) Average packet delivery ratio

FIGURE 11. Performance behavior under blackhole attack in the 1^{st} scenario.

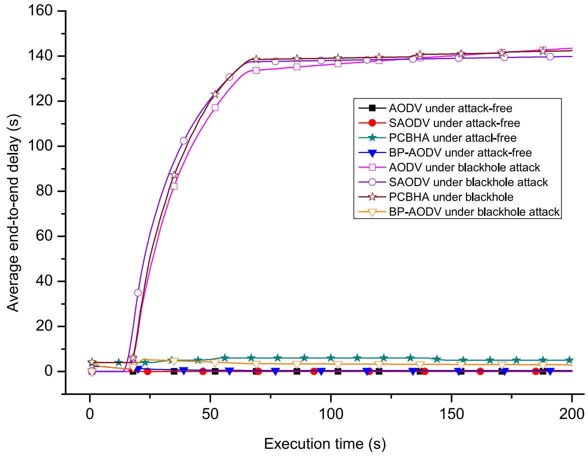


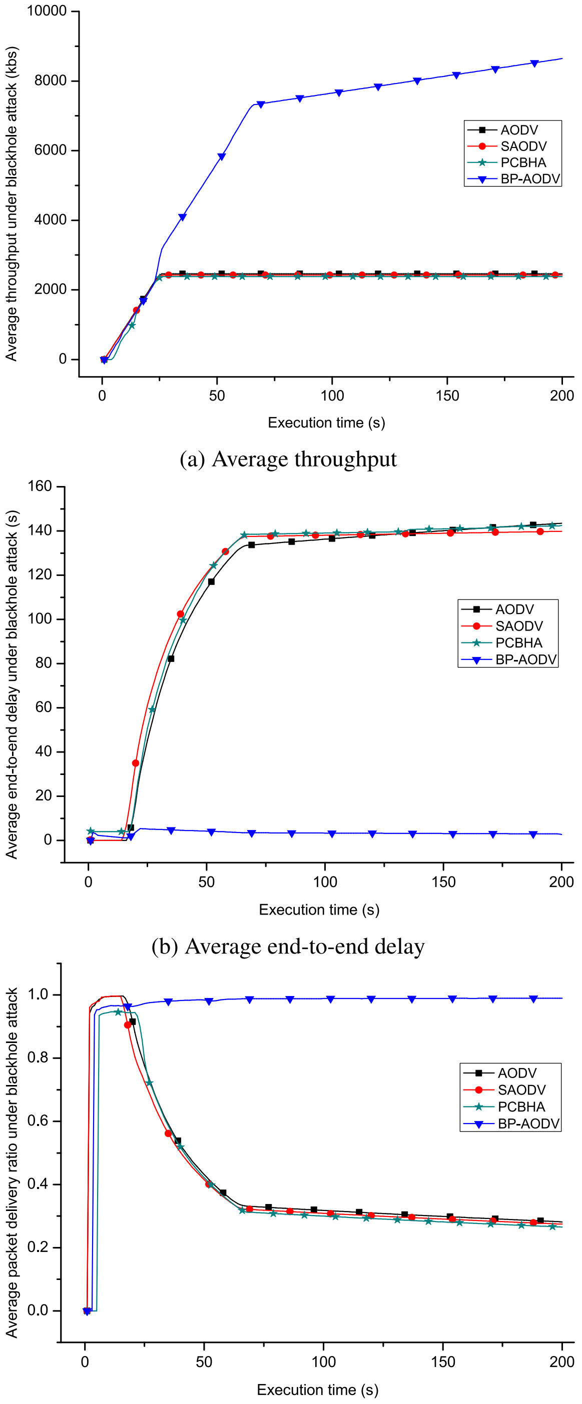
FIGURE 12. Average end-to-end delay.

vulnerable under this attack. The results also reveal that employing two malicious nodes can violate the security of the SAODV against blackhole attack protection. As discussed in Section III-C, the two malicious nodes are cooperated to successfully attach one of them on the route paths of *con*2 and *con*3 during the routing process. Next, when one of the malicious nodes receives any data packet via *con*2 or *con*3, it simply drops the packet. Also the PCBHA is vulnerable to the cooperative blackhole attack when a malicious node forges the ACK packets to increase its fidelity or trust level and lure the source node to create a route through one of the malicious nodes that drops the data packets during the forwarding process. As a result, the performance of SAODV and PCBHA protocols are degraded as visualized in Fig. 13. This means that the average throughput and PDR are decreased while the average end-to-end delay is increased during each of the connections *con*2 and *con*3. Note that Even though the PCBHA protocol protects against the cooperative blackhole attack, its security can be broken by forging the acknowledgement packets as shown in the results of the first and second scenarios.

During the third scenario, also two experiments are carried out to measure the average throughput, average end-to-end delay, and average PDR of the four protocols. The experiments also use one second as step size to calculate the averages. The first experiment evaluates the average throughput behavior under attack-free condition. The result in Fig. 14 demonstrates that the average throughput of each of the four protocols exhibits almost the same behavior.

On the other hand, the second experiment of the third scenario measures the performance of the four protocols under the blackhole attack generated by a malicious node that behaves normally during the routing process to avoid the detection but if it is attached to a route path, it drops any received data packets during the forwarding process. The experiment uses node 3 in Fig. 9 as a malicious node and its results are visualized in Fig. 15.

The results of the average throughput in Fig. 13a, end-to-end delay in Fig. 13b, and PDR in Fig. 13c assure that the malicious node is attached to some of the route paths established by the AODV and SAODA protocols while it is avoided by the PCBHA and BP-AODV protocols. Specifically, the results show that the malicious node is attached to connections *con*2 and *con*3 created by the AODV protocol while it is attached to the connection *con*3 established by the SAODV protocol. Since the AODV and SAODV protocols



(a) Average throughput

(b) Average end-to-end delay

(c) Average packet delivery ratio

FIGURE 13. Performance behavior in the 2nd scenario under cooperative blackhole attack.

cannot detect and avoid such malicious node during the forwarding process, their performance will go down. As a result, the average throughput in Fig. 13a and PDR in Fig. 13c obtained by the AODV start degrading during *con*2 and *con*3. In contrast, the average throughput and PDR produced by the SAODV protocol start to go down during *con*3. In the same way, the average end-to-end delay in Fig. 13b resulted from the AODV protocol goes up during *con*2 and *con*3 while the average end-to-end delay produced from the SAODV protocols starts increasing during *con*3.

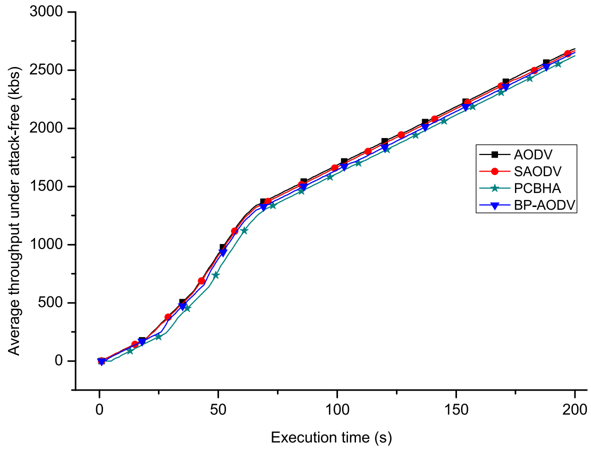
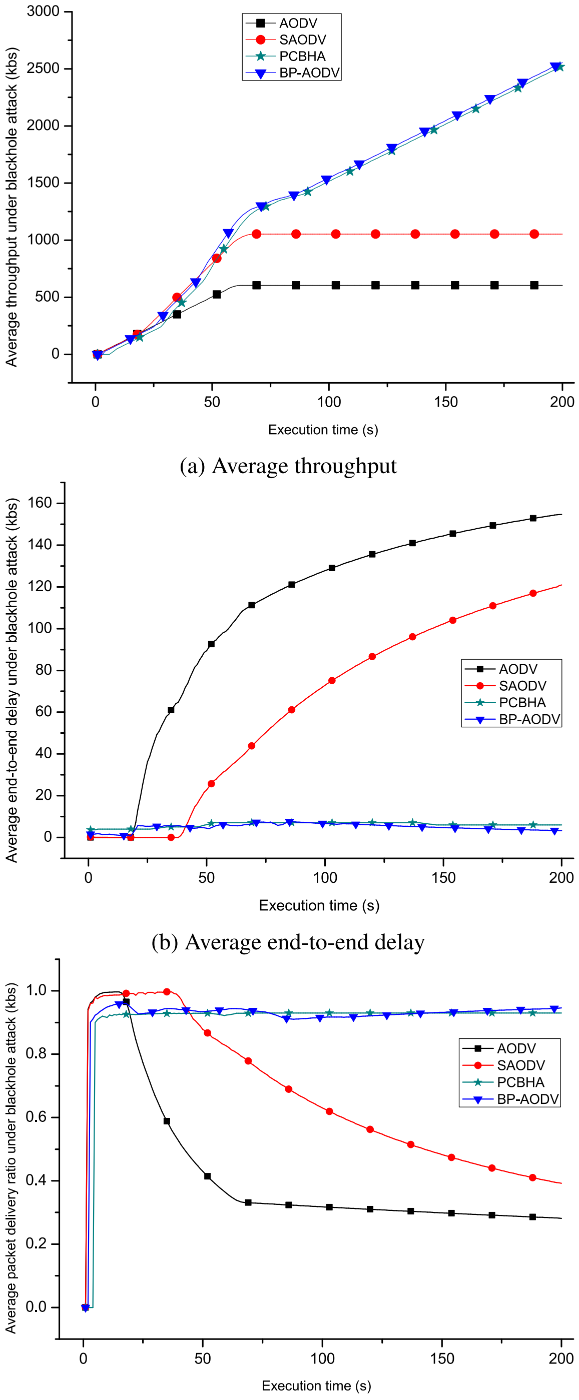


FIGURE 14. The average throughput under attack-free in the 3rd scenario.

On the other hand, the PCBHA and BP-AODV protocols are able to detect and avoid such malicious node during the forwarding process. During this scenario, the malicious node in the PCBHA protocol drops any received data packets during the forwarding process without forging any acknowledgement packets. Since the malicious node does not forge ACK packets, its fidelity or trust level will be reduced. Reducing the fidelity level of the malicious node helps the source node to detect the malicious node and avoid creating a route through it during the routing process. As a result, the PCBHA protects against the blackhole attack during the underlying scenario. The BP-AODV protocol considered such case during its design as discussed in Section IV-C. When the BP-AODV protocol detects the malicious node on a route path during the forwarding process, the protocol avoids this route by choosing another route to forward data packets through it. Note that the BP-AODV protocol establishes up to three routes for each connection. Consequently, if a route has a malicious node, the BP-AODV protocol is able to choose another route based on Equation 6 without the need to establish the connection again. The results produced by the PCBHA and BP-AODV protocol in Fig. 15 assure their robustness against a malicious node that behaves normally during the routing process while dropping their received data packets during the forwarding process. Specifically, the average throughput in Fig. 13a and PDR in Fig. 13c obtained by each of the PCBHA and BP-AODV protocol is very high while the average end-to-end delay in Fig. 13b is very low compared with the AODV and SAODV protocols.

From the results obtained in Fig. 10 to Fig. 15, we can conclude that the AODV has a very good performance under attack-free while it experiences a very poor performance during blackhole attack performed in the three mentioned scenarios. Also the SAODV protocol performs very well under both attack-free and a blackhole attack generated using one malicious node during the first scenario. On the other



(a) Average throughput

(b) Average end-to-end delay

(c) Average packet delivery ratio

FIGURE 15. Performance behavior under blackhole attack of the 3^{rd} scenario.

hand, the SAODV protocol produces very poor performance during the cooperative blackhole attack in the second scenario and the blackhole attack of the third scenario. The results also show that the PCBHA protocol is vulnerable to the blackhole attack when the malicious node forges the ACK packets as depicted in Fig. 11, Fig. 12, and Fig 13. On the other hand, when the malicious node does not forge the ACK packet as in the third scenario, it will be detected and avoided by the source node as revealed by the results in Fig. 15. Finally, the results approve that the BP-AODV protocol exhibits very good performance during attack-free and blackhole attack during the three underlying scenarios.

VI. CONCLUSION

The paper introduced a blackhole protected ad-hoc on demand distance vector (BP-AODV) routing protocol for MANETs. The BP-AODV addressed the blackhole vulnerability associated with each of the AODV and SAODV protocols. In addition, it utilized the chaotic map features to protect against cooperative blackhole attack that is performed by two malicious nodes. Furthermore, the BP-AODV considered and protected against the blackhole attack that might be achieved by a malicious node that behaves normally during the routing process but maliciously during the forwarding process. The BP-AODV is implemented using the well-known network simulator version 2 (NS2) and compared against the AODV, SAODV, and PCBHA protocols. The experimental results showed that the BP-AODV protocol is effective in thwarting blackhole attack that might be occurred in different scenarios.

REFERENCES

[1] A. A. Hanafy, S. H. Noureldin, and M. A. Azer, "Immunizing the SAODV protocol against routing information disclosure," in *Proc. Int. Conf. Internet Technol. Secured Trans.*, Abu Dhabi, United Arab Emirates, Dec. 2011, pp. 330–334.

[2] S. Prakash and A. Swaroop, "A brief survey of blackhole detection and avoidance for ZRP protocol in MANETs," in *Proc. Int. Conf. Comput., Commun. Automat. (ICCCA)*, Noida, India, Apr. 2016, pp. 651–654.

[3] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *Proc. Conf. Commun. Archit., Protocols Appl.*, London, U.K., Aug./Sep. 1994, pp. 234–244.

[4] Y. Ye, S. Feng, M. Liu, X. Sun, T. Xu, and X. Ting, "A safe proactive routing protocol SDSDV for ad hoc network," *Int. J. Wireless Inf. Netw.*, vol. 25, no. 3, pp. 348–357, 2018.

[5] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, "Optimized link state routing protocol for ad hoc networks," in *Proc. IEEE Int. Multi Topic Conf.*, Lahore, Pakistan, Dec. 2001, pp. 62–68.

[6] Z. Wang, Y. Chen, and C. Li, "PSR: A lightweight proactive source routing protocol for mobile ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 2, pp. 859–868, Feb. 2014.

[7] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing* (International Series in Engineering and Computer Science), vol. 353, T. Imielinski and H. Korth, Eds. Boston, MA, USA: Springer, 1996, pp. 153–181.

[8] J. Liu, F. Fu, J. Xiao, and Y. Lu, "Secure routing for mobile ad hoc networks," in *Proc. 8th ACIS Int. Conf. Softw. Eng., Artif. Intell., Netw., Parallel/Distrib. Comput.*, Qingdao, China, Jul./Aug. 2007, pp. 314–318.

[9] C. E. Perkins, and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. 2nd IEEE Workshop Mobile Comput. Syst. Appl.*, New Orleans, LA, USA, Feb. 1999, pp. 90–100.

[10] S. Lu, L. Li, K.-Y. Lam, and L. Jia, "SAODV: A MANET routing protocol that can withstand black hole attack," in *Proc. Int. Conf. Comput. Intell. Secur.*, Beijing, China, Dec. 2009, pp. 421–425.

[11] Z. Haas, M. Pearlman, and P. Samar, *The Zone Routing Protocol (ZRP) for Ad Hoc Networks*, Internet Draft, IETF MANET Working Group, Jul. 2002. [Online]. Available: http://www.ietf.org/proceedings/02nov/I-D/draft-ietf-manet-zone-zrp-04.txt

[12] D. Ravilla, V. Sumalatha, and C. Putta, "Hybrid routing protocols for ad hoc wireless networks," *Int. J. Ad hoc, Sensor Ubiquitous Comput.*, vol. 2, no. 4, pp. 79–96, 2011.

[13] S. Kalita, B. Sharma, and U. Sharma, "Attacks and countermeasures in mobile ad hoc network—An analysis," *Int. J. Adv. Comput. Theory Eng.*, vol. 4, no. 3, pp. 16–21, 2015.

[14] H. L. Nguyen and U. T. Nguyen, "A study of different types of attacks on multicast in mobile ad hoc networks," *Ad Hoc Netw.*, vol. 6, no. 1, pp. 32–46, 2008.

[15] M. C. Trivedi, S. Yadav, and V. K. Singh, "Securing ZRP routing protocol against DDoS attack in mobile ad hoc network," in *Advances in Data and Information Sciences* (Lecture Notes in Networks and Systems), vol. 39, M. L. Kolhe, M. C. Trivedi, S. Tiwari, and V. K. Singh, Eds. Singapore: Springer, 2019, pp. 387–396.

[16] M. Hussain and A. Devaraj, "Upshot of sinkhole attack in DSR routing protocol based MANET," *Int. J. Eng. Res. Appl.*, vol. 3, no. 2, pp. 1737–1741, 2013.

[17] M. J. Faghihniya, S. M. Hosseini, and M. Tahmasebi, "Security upgrade against RREQ flooding attack by using balance index on vehicular ad hoc network," *Wireless Netw.*, vol. 23, no. 6, pp. 1863–1874, 2017.

[18] C. Panos, C. Ntantogian, S. Malliaros, and C. Xenakis, "Analyzing, quantifying, and detecting the blackhole attack in infrastructure-less networks," *Comput. Netw.*, vol. 113, pp. 94–110, Feb. 2017.

[19] N. Khanna and M. Sachdeva, "A comprehensive taxonomy of schemes to detect and mitigate blackhole attack and its variants in MANETs," *Comput. Sci. Rev.*, vol. 32, pp. 24–44, May 2019.

[20] L. Mejaele and E. O. Ochola, "Analysing the impact of black hole attack on DSR-based MANET: The hidden network destructor," in *Proc. 2nd Int. Conf. Inf. Secur. Cyber Forensics*, Cape Town, South Africa, Nov. 2015, pp. 140–144.

[21] T. V. Thong and L. Buttyán, "On automating the verification of secure ad-hoc network routing protocols," *Telecommun. Syst.*, vol. 52, no. 4, pp. 2611–2635, 2013.

[22] L. Tamilselvan and V. Sankaranarayanan, "Prevention of co-operative black hole attack in MANET," *J. Netw.*, vol. 3, no. 5, pp. 13–20, 2008.

[23] E. O. Ochola, L. F. Mejaele, M. M. Eloff, and J. A. van der Poll, "Manet reactive routing protocols node mobility variation effect in analysing the impact of black hole attack," *SAIEE Afr. Res. J.*, vol. 108, no. 2, pp. 80–92, Jun. 2017.

[24] A. K. Jain and A. Choorasiya, "Security enhancement of AODV routing protocol in mobile ad hoc network," in *Proc. 2nd Int. Conf. Commun. Electron. Syst. (ICCES)*, Coimbatore, India, Oct. 2017, pp. 958–964.

[25] M. Medadian, A. Mebadi, and E. Shahri, "Combat with black hole attack in AODV routing protocol," in *Proc. IEEE 9th Malaysia Int. Conf. Commun. (MICC)*, Kuala Lumpur, Malaysia, Dec. 2009, pp. 530–535.

[26] S. Dokurer, Y. M. Erten, and C. E. Acar, "Performance analysis of ad-hoc networks under black hole attacks," in *Proc. IEEE SoutheastCon*, Richmond, VA, USA, Mar. 2007, pp. 148–153.

[27] L. Tamilselvan and V. Sankaranarayanan, "Prevention of blackhole attack in MANET," in *Proc. 2nd Int. Conf. Wireless Broadband Ultra Wideband Commun.*, Sydney, NSW, Australia, Aug. 2007, p. 21.

[28] R. C. Debarati, R. Leena, and M. Nilesh, "Implementing and improving the performance of AODV by receive reply method and securing it from black hole attack," *Procedia Comput. Sci.*, vol. 45, pp. 564–570, Mar. 2015.

[29] A. A. Chavan, D. S. Kurule, and P. U. Dere, "Performance analysis of AODV and DSDV routing protocol in MANET and modifications in AODV against black hole attack," *Procedia Comput. Sci.*, vol. 79, pp. 835–844, 2016.

[30] S. R. Deshmukh, P. N. Chatur, and N. B. Bhople, "AODV-based secure routing against blackhole attack in MANET," in *Proc. IEEE Int. Conf. Recent Trends Electron., Inf. Commun. Technol.*, Bengaluru, India, May 2016, pp. 1960–1964.

[31] A. Yasin and M. A. Zant, "Detecting and isolating black-hole attacks in MANET using timer based baited technique," *Wireless Commun. Mobile Comput.*, vol. 2018, Sep. 2018, Art. no. 9812135.

[32] H. Kaur and K. Mangat, "Black hole attack in mobile ad hoc networks: A review," *Int. J. Advance Res., Ideas Innov. Technol.*, vol. 3, no. 2, pp. 189–191, 2017.

[33] H. Yih-Chun and A. Perrig, "A survey of secure wireless ad hoc routing," *IEEE Security Privacy*, vol. 2, no. 3, pp. 28–39, May 2004.

[34] S. Ramaswamy, H. Fu, M. Sreekantaradhya, J. Dixon, and K. Nygard, "Prevention of cooperative black hole attack in wireless ad hoc networks," in *Proc. Int. Conf. Wireless Netw.*, Las Vegas, NV, USA, 2003, pp. 570–575.

[35] C. Hongsong, J. Zhenzhou, and H. Mingzeng, "A novel security agent scheme for AODV routing protocol based on thread state transition," *Asian J. Inf. Technol.*, vol. 5, no. 1, pp. 54–60, 2006.

[36] C. Perkins, E. Belding-Royer, and S. Das, *Ad Hoc on Demand Distance Vector (AODV) Routing*, document RFC 3561, IETF MANET Working Group, 2003, pp. 570–575. [Online]. Available: http://www.ietf.org/rfc/rfc3561.txt

[37] S. Strogatz, *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology Chemistry, and Engineering*, 2nd ed. Boulder, CO, USA: Westview, 2015.

[38] M. Wang, X. Wang, Y. Zhang, and Z. Gao, "A novel chaotic encryption scheme based on image segmentation and multiple diffusion models," *Opt. Laser Technol.*, vol. 108, pp. 558–573, Dec. 2018.

[39] K. S. Praveen, H. L. Gururaj, and B. Ramesh, "Comparative analysis of black hole attack in ad hoc network using AODV and OLSR protocols," *Procedia Comput. Sci.*, vol. 85, pp. 325–330, Jan. 2016.

[40] M. A. Abdelshafy and P. J. B. King, "Resisting blackhole attacks on MANETs," in *Proc. 13th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Las Vegas, NV, USA, Jan. 2016, pp. 1048–1053.

[41] T. Issariyakul and E. Hossain, *Introduction to Network Simulator NS2*. Boston, MA, USA: Springer, 2009. [Online]. Available: https://link.springer.com/chapter/10.1007/978-0-387-71760-9_2

ALY M. EL-SEMARY was born in Egypt, in 1969. He received the B.S. degree in systems and computer engineering from Al-Azhar University, Cairo, Egypt, in 1992, and the M.S. and Ph.D. degrees in network security from Tulsa University, OK, USA, in 2001 and 2004, respectively.

Since 1994, he has been a Teaching Assistant with the Department of Systems and Computer Engineering, Faculty of Engineering, Al-Azhar University, where he is currently an Associate Professor. He is also as a Visitor with the Computer Engineering Department, Collage of Computer Science and Engineering, Taibah University, Saudi Arabia. He is the author of several articles, conference papers, and book chapters. His current interests include network and computer security, wireless and sensor networks, application of chaotic systems in multimedia encryption, digital image processing, data mining, and neural networks. In 1998, he received the Full Scholarship from the Government of Egypt to study his M.S. and Ph.D. degrees at Tulsa University. He is also a Reviewer for several journals and conferences.

HOSSAM DIAB was born in Egypt, in 1978. He received the B.S., M.Sc., and Ph.D. degrees in computer science from the Faculty of Science, Menoufia University, Egypt, in 1999, 2004, and 2010, respectively.

He is currently an Associate Professor with the Department of Mathematics and Computer Science, Faculty of Science, Menoufia University. He is also a Visitor with the Computer Science and Engineering College, Taibah University, Saudi Arabia. His research interests include the areas of cryptography, network security, application of chaotic systems in multimedia content encryption, digital image processing, image compression, and image watermarking.