PREVENTIVE MECHANISM AGAINST SIMPLE AND COOPERATIVE BLACK HOLE ATTACKS IN MULTI-HOP WIRELESS AD HOC NETWORKS

Praveenkumar.R[1]
PG Student ,
Department of Computer applications,
IFET College of Engineering,
Villupuram.

SivaSankaran.S[2]
Senior Assistant Professor,
Department of Computer application,
IFET College of Engineering,
Villupuram.

Abstract

This paper focuses on the issues related to the simple and cooperative black hole attacks in multi-hop wireless ad hoc networks. In multi-hop wireless ad hoc networks, nodes not in direct range rely on intermediate nodes to communicate. In order to preserve its limited resources or to launch a DoS attack, an intermediate node drops packets going through it instead to forward them to its succes- sor. In this paper, we deal with this misbehavior called black hole attack, and we propose an authenticated end-to-end acknowledgment based approach in order to check the correct forwarding of packets by intermediate nodes. Our approach detects the black hole conducted in simple or cooperative manner, the modification and the replay of messages attacks. Through simulation using OPNET simulator, we show the detection efficiency and evaluate the performance of our approach in both proactive and reactive routing based networks in terms of end-to-end delay and network load. Also, we compare the approach we propose with 2-hop ACK and the watchdog approaches in terms of detection ratio, delivery ratio and additional overhead.

KEYWORDS:

Wireless ad hoc network, Routing protocol security, Simple black hole, Cooperative black hole, OPNET simulator, DoS attack, 2-hop ACK.

I. INTRODUCTION

A multi-hop wireless ad hoc network is a collection of autonomous nodes that communicate with each other by forming a multi-hop radio network. It can be easily and rapidly deployed without the aid of any established infra- structure or centralized administration. Such network has some special features such as open and unreliable wireless links, constantly changing network topologies and memberships, limited bandwidth, battery, lifetime, and computation power of nodes. While these features make the network more flexible, they introduce specific security concerns. Indeed, an ad hoc network is vulnerable to various types of attacks including passive eavesdropping, impersonation, and denial-of-service. Preventive or detective security measures using cryptographic tools such as digital signature, public key encryption, and non-crypto- graphic tools such as Intrusion Detection System (IDS), can improve the security of the network. However, these tech- niques can address only a subset of the threats, and the problem remains always open and the remedy is far from being obvious.

II. OBJECTIVE

The scope of the project is to avoid packet dropping to intermediate nodes, In multi-hop wireless ad hoc networks, the cooperation amongst nodes is essential to deliver packets to the desti- nation node. An intermediate node, that participates vol- untarily in routing and packets forwarding operations, can behave selfishly or maliciously to drop packets going through it, instead of forwarding them to its successor. The dropper aim is the preservation of its resources like its limited energy (selfish behavior), or the launch of denial of service attack (malicious behavior). This misbehavior, called black hole attack, can be conducted by one interme- diate node (simple black hole) or results on the coopera- tion of several intermediate nodes (cooperative black hole). In this paper, we propose an end-to-end authenticated ACK based approach to check the correct forwarding of packets by intermediate nodes. Our main goal is the detection of simple and coop- erative black hole attacks, and as a secondary objective, we detect the modification and the replay of messages. We note that the modification and the replay of messages, completely ignored in existing approaches, are essential to deliver packets to the destination node. Through simula- tion we show the detection efficiency and evaluate the

performance of our approach in terms of end-to-end delay and network load in both AODV and OLSR based networks.

III. PROBLEM STATEMENT

To cope with this attack, existing approaches are mainly based on monitoring individual nodes,and they focus on the black hole conducted in single or cooperative manner, but not both simultaneously. The black hole attack causes a serious damage on the network.The authors provided a simulation study in which an AODV-based network performance, in the pres- ence of black hole nodes, is reduced up to 26%. To cope with this attack, researchers proposed solutions against black hole attack acting in an individual or a cooperative manner, or they proposed security mechanisms to cope with other attacks additionally to the black hole attack.

IV. PROPOSED SOLUTION

The proposed system supports both single or cooperative manner simultaneously.we propose a routing security protocol, where the intermediate node sends back to the source its next hop information with the reply. To verify whether the next hop has a link with the intermediate node, the source sends a further request packet to the next hop. The latter should send back a further reply message which includes the check result. If the next hop ensures that the intermediate node exists, the source starts to establish a route to the destination through this intermediate node. This protocol generates an important overhead due to fur- ther request and reply packets. Cooperative black hole is when several malicious nodes work together as a group. To identify multiple black hole nodes acting in cooperation, authors in propose a slight modification in AODV, where a Data Routing Infor- mation (DRI) table is used to save information on routing data packet from/through the node. The DRI helps to deter- mine reliable nodes used to discover secure paths from source to destination.

A.BLACK HOLE ATTACK MODEL

The simple blackhole is when the dropper acts individ- ually to carry out its attack. First, the dropper violates the routing protocol specification to advertise itself as having a valid route to a destination node, then it drops the intercepted packets without forwarding them. In order to clarify this attack, we will describe, in the following sub- sections, how the simple black hole is conducted in AODV and OLSR routing protocols.

B.BLACK HOLE ATTACK MODEL IN AODV

Ad hoc On-Demand Distance Vector (AODV) [18] is a reactive routing protocol designed to provide routes on demand. In AODV, the source node broadcasts a route request (RREQ) message containing amongst other infor- mation: destination's address, destination's sequence number, hop counter. Neighbors of the source node update their routing tables accordingly and broadcast RREQ. This process is reiterated until RREQ reaches the destination node. Thus, the latter uses the preestablished reverse route to send back a route reply (RREP) to the source node. It should be noted that the source node can receive several RREPs from different nodes. However, it chooses the one with greater sequence number for the intended destina- tion. If RREPs containing highest sequence number for the same destination are reported by more than one node, then the path with smaller hop counter will be selected. In this regard, the black hole attack in AODV is summarized in the following points:

1. When the black node receives a RREQ, it takes note of the destination address, and prepares a RREP, in which the destination address is set to the spoofed destination address, the sequence number is set to a greatest value and the hop counter is set to a small- est value.

2. The black hole node sends RREP to the closest inter- mediate node belonging to the actual active route.

3. RREP received by the intermediate node will be relayed through the reverse path towards the source node.

4. The source node updates its routing table according to the received RREP, and uses the new route to send data.

5. When intercepted, data will be dropped the black hole node.

C.BLACK HOLE ATTACK MODEL IN OLSR

Optimized Link State Routing protocol (OLSR) is a proactive routing protocol designed to provide routes immediately when needed. In OLSR, MPR (Multi-Point Relay) nodes play a paramount role in the network, because they forward broadcast messages during the flooding process. The MPR set of nodes is computed such that a broadcast message, retransmitted by these selected MPR nodes, will be received by all nodes 2-hops away. The information required to perform the calculation of the MPR set is acquired through the periodic exchange of HELLO messages. Among fields of a HELLO message, the Willingness field that specifies the willingness of a node to carry and forward traffic for other nodes. According to OLSR specification, a node with willingness field set to WILL_ALWAYS must always be selected as MPR. Therefore, to conduct a black hole attack, a malicious node must constantly maintain its Willingness field to WILL_ALWAYS in all its disseminated HELLO messages. Given this, a black hole node forces its election as MPR. Once elected, it will belong certainly to the end-to-end route. This way, it will drop all or selected messages that pass through it. The following points summarize the black hole attack in OLSR:

1. The black hole node prepares HELLO message in which the field Willingness is set to WILL_ALWAYS, and broadcasts it to its neighbors.

2. Neighbors elect the black hole node as MPR, and build their routing tables based on it.

3. Any node whose routing table contains black hole node may become the victim of the black hole.

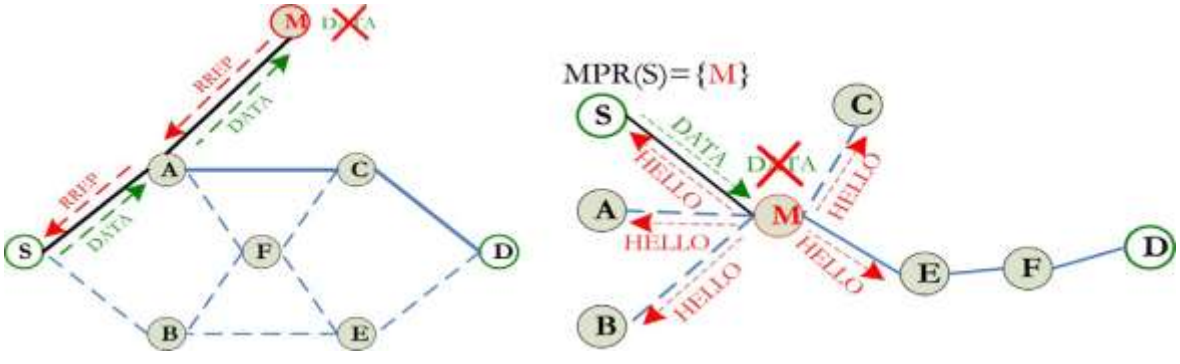


Fig 1. (a) AODV-based network **(b) OLSR-based network**

D.COOPERATIVE BLACK HOLE

In the cooperative black hole, multiple blackhole nodes act in coordination to violate the routing protocol specification or the implemented security mechanism. In order to clarify this attack, the following paragraph describes the situation where multiple blackhole nodes act in coordination to violate the security mechanism, for example, to identify a simple black hole node.

As depicted in Fig. 2, when blackhole nodes M1 and M2 act together, M1 refers to M2 as its next hop. In the security mechanism proposed in the source node S sends a further request packet (FReq) to M2 through another route (S; C; E; M2) other than via M1 . The node S asks M2 if it is the next hop of M1 and if it has a valid route to the

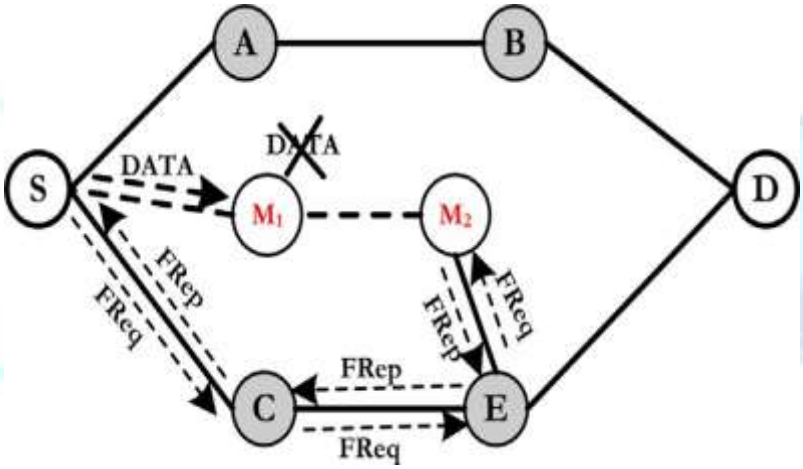


Fig. 2. Cooperative black hole attack.

V. OUR PROPOSAL

In this section, we introduce the network assumptions and detail our security approach against the simple and the cooperative black hole attacks.

V.1. ASSUMPTIONS

We assume that wireless links are bidirectional, because our solution requires a bidirectional exchange of packets. We suppose also that the source nœud n0 shares a common key ki with intermediate nodes nj , 1 < j < m, where m is the number of nodes in the

end-to-end path. Additionally, we suppose that the source node n0 trusts the destination node nm 1 , i.e. the destination node does not disclose in any case its shared key. We note that these assumptions are all reasonable and practically realizable.

V.II. SOLUTION OVERVIEW

In order to facilitate the comprehension of our approach, we will initially describe it for three nodes, then we extend it to a general scenario of several nodes.Let A; B and C be three successive nodes and msg is a message to be sent from A to C via B. Our solution must ensure the following points:

1. C must acknowledge msg, i.e. C must confirm the reception of msg.
2. B must be prevented to replay the role of C, i.e. B cannot send messages to A by impersonating C.
3. B will not modify messages passing through it.
4. Two nodes in the end-to-end path cannot cooper- ate to lead the attack.

To implement our solution, we use the common key k shared between A and C. This key is used to encrypt mes- sages sent by A to C (or sent by C to A) via B, and to decrypt messages received by C from A (or received by A from C) via B. We use also a hash function h in order to ensure the integrity of messages passing through a potential attacker B. An example of such hash function can be SHA1 . Additionally, we use a bijective function f, i.e. for all

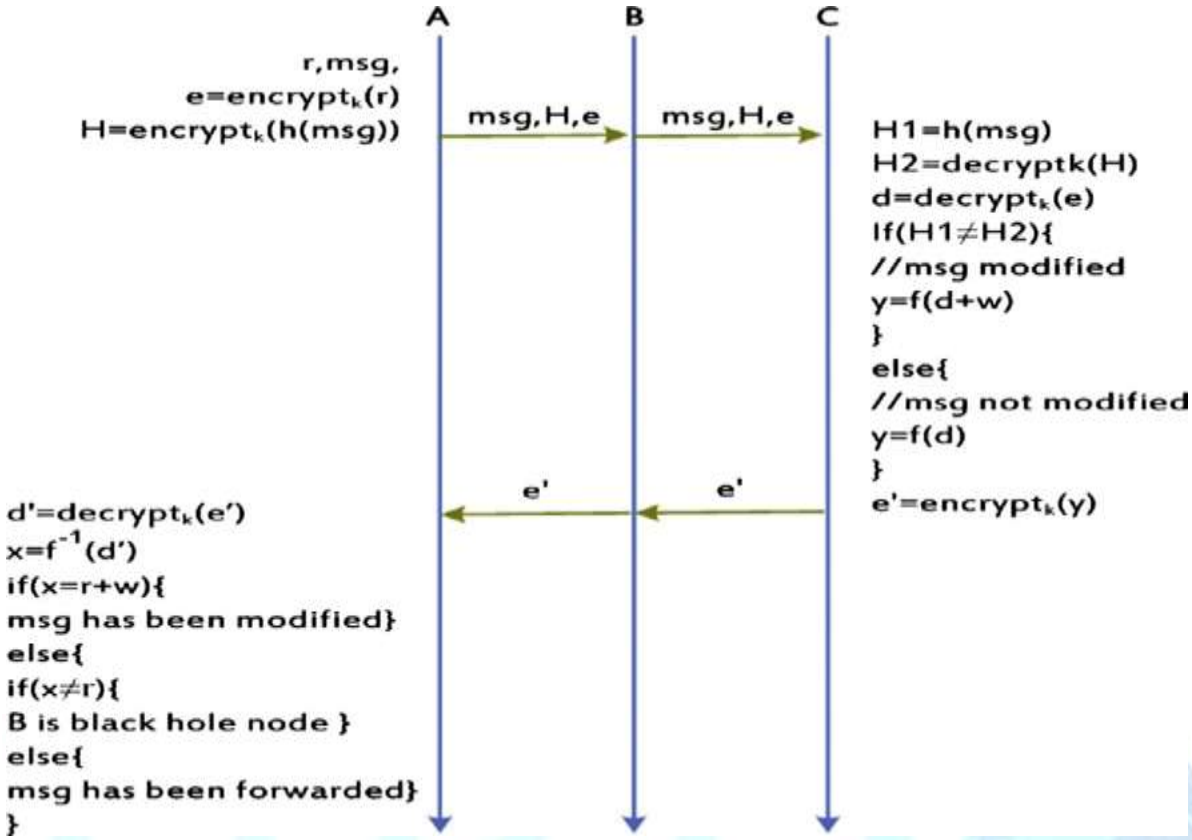


Fig. 3. Detection of black hole attack, case of 3 nodes.

y ¼ f ðxÞ, there exists a unique value x such that x ¼ f 1 ðyÞ. An example of such bijective function can be y ¼ f ðxÞ ¼ x þ 1. The function f is used to prevent the replay of A's messages by B. Finally, we utilize the constant value w used by C, if necessary, to inform A that the mes- sage was modified by B.As illustrated in Fig. 3, before sending a message msg to C, the node A generates a random value r and encrypts it in order to obtain the value e, i.e. e ¼ encryptk ðrÞ. Then, A computes the hash hðmsgÞ, encrypts hðmsgÞ using k(H ¼ encryptk ðhðmsgÞÞ) and sends the triplet: (msg; H; e) to C via B. When the triplet is received by C, the latter ver- ifies the integrity of msg by recomputing H1 ¼ hðmsgÞ, decrypts H using k (H2 ¼ decryptk ðHÞ) and compares H2 to H1 . If H2 – H1 then msg has been modified. The node C obtains d by decrypting e using k and computes y, where y ¼ f ðdÞ if msg is not modified or y ¼ f ðd þ wÞ otherwise. C encrypts y in order to obtain e0 , then it sends back e0 to A via B. When e0 is received by A, this latter gets d0 byback e0 ¼ encryptk ðdecryptk ðeÞÞ instead of e0 ¼ encryptk ðf ðdecryptk ðeÞÞÞ to A via B. The latter will intercept e and send back it to A by impersonating C. B cannot perform

this with e0 ¼ encryptk ðf ðdecryptk ðeÞÞ because e and e0 are differ- ent in this case. Therefore, B cannot replay C's message if the function f is used. Obviously, the message msg cannot be modified by B, because the encrypted hash H is attached to it.

ALGORITHM 1. CHECK_REC(MSG; H; E)

1: ej extractðEÞ

2: dj decryptkj (ej)

3: hj extractðHÞ

4: H2 decryptk

5: H1 hðmsgÞ

6: if ðH1 – H2 Þ then

7: The message msg was modified by the node nj 1
instead of C. If x ¼ r then msg was well forwarded by B.

8: e0

9: else
10: e0

11: if (j < m 1) then

12: E ¼ E ej

13: H ¼ H hj

14: Send(njþ1 ; E; msg; H)

15: end if

16: end if

17: Send(n0 ; e0)

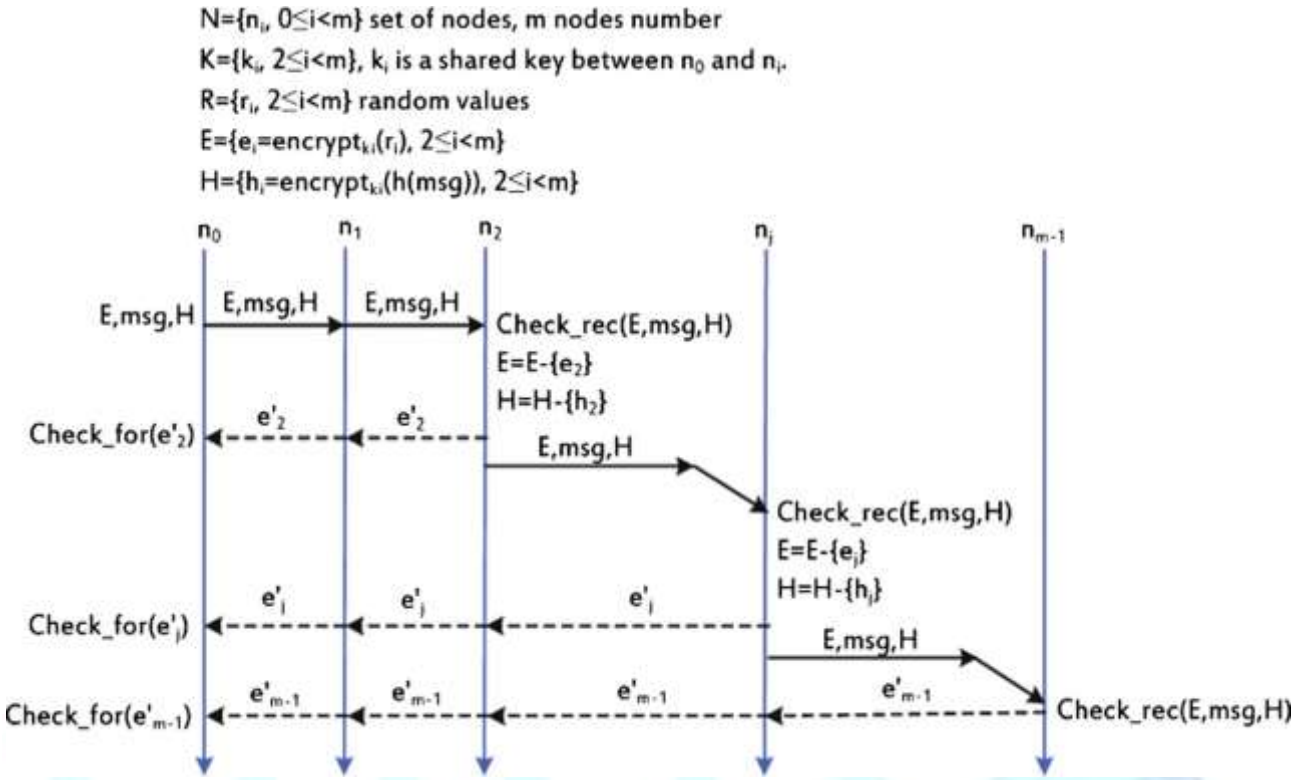


Fig. 4. Detection of black hole attack, general scenario.

Let e0 be equal to encrypt(f ðdj þ wÞ) or encryptkj (f ðdj Þ) according to whether msg is modified or not. nj sends e0back to n0 to inform it that msg was well received or hasbeen modified by nj 1 . In addition, nj raises its information ej and hj from E and H respectively, then it forwards the triplet (msg; H; E) to the next node njþ1 ðexceptj ¼ m 1 which is the destination node). Note that the node nj for- wards the triplet (msg; H; E) to njþ1 if and only if the mes- sage msg was successfully received.When n0 receives e0 , it executes Algorithm 2, where n0 decrypts e0 using kj in order to obtain d0 , and computesxj ¼ f ðdj Þ. Based on xj ; n0 determines the status of msg.If xj ¼ ðrj þ w) then msg was modified by nj 1 ; if xj – rj then nj 1 was not forwarded msg and it is a black hole node; if xj ¼ rj then nj 1 was well forwarded msg. Note that the node n0 must receive e0 from the node nj before atime-out tj . If this time-out is exceeded, the node nj 1 is suspected to be dishonest node. If all checks of acknowl- edgments e0 have been carried out successfully, the mes- sage msg has been properly forwarded to the destination node.

ALGORITHM 2. CHECK_FOR(E0)

1: d0decryptkj(e0)

2: xj f 1 ðd0 Þ

3: if ðxj ¼ rj þ wÞ then

4: The message msg was modified by the node nj 1 .

5: else

6: if (xj – rj) then

7: the node nj 1 was not forwarded msg and it is black hole one.

8: else

9: The message was forwarded by nj 1

10: end if

11: end if

VI.BLACK HOLE ATTACK DETECTION

The purpose of the black hole attack is to prevent traffic from reaching its destination. For this reason, we have measured the following metrics:

1. Traffic sent by the source node (packet/s): denotes the number of packets/s sent by the source node.

TABLE 1

SIMULATION PARAMETERS.

Simulation parameter	Value
Number of nodes	20
Network size	1 km／1 km
Simulation duration	400 s
Traffic generation start time	20 s
Packet inter-arrival time (s)	Exponential (1)
Packet size (bits) Transmit power (W)	Exponential (1024)
Mobility model	0.001
Routing protocol	Random way point
Hash function	AODV, OLSR SHA-1

2. Traffic received by the destination node (packet/s): denotes the number of packets/s received by the destination node.

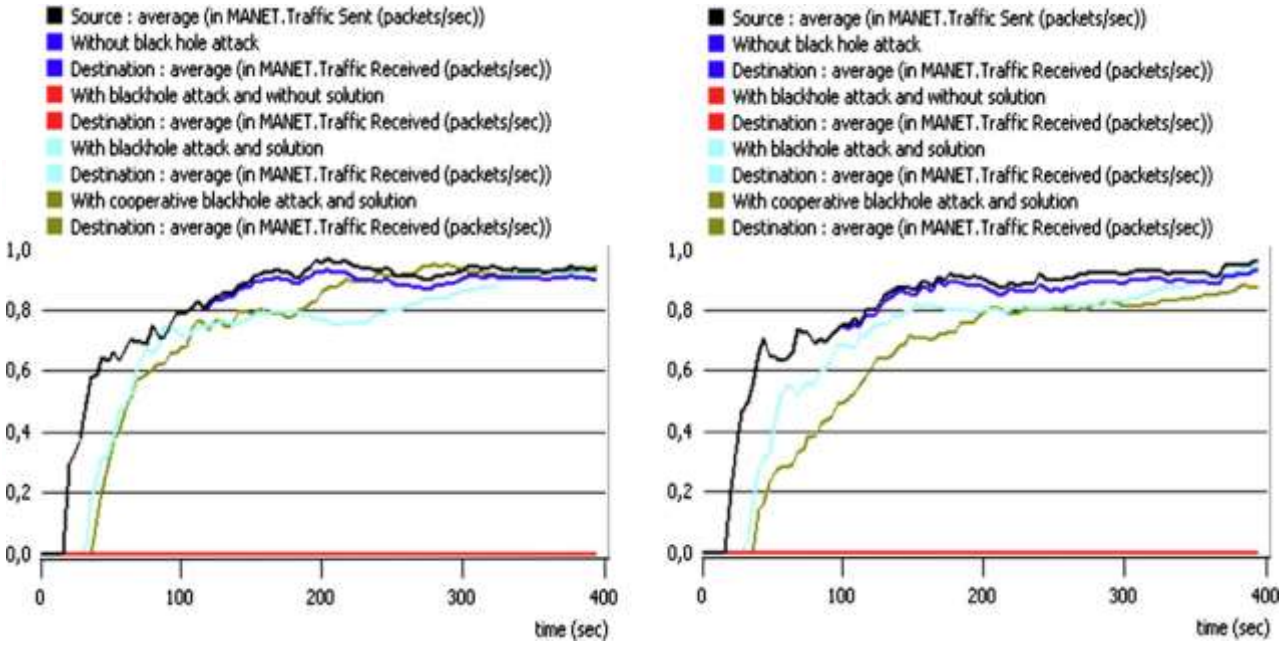
VII.PERFORMANCE EVALUATION

Intuitively, if the end-to-end delay and the network load are small, then the network is more efficient. So, in order to evaluate the network performance, we have mea-sured the two following metrics:

1. End-to-end delay: denotes the delay, in second, to send a bit from the source to the destination.
2. Network load: denotes the traffic quantity, in bits/s, in the entire network.

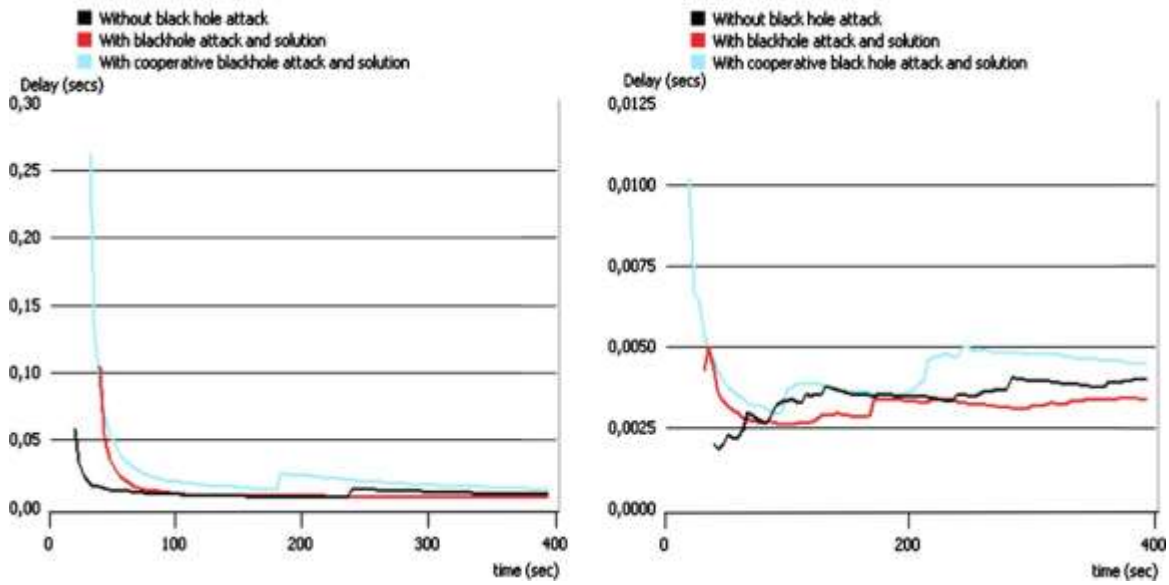
Fig. 6 shows the end-to-end delay in both AODV-based network (Fig. 6)) and OLSR-based network (Fig. 6b), in cases: network without black hole; network with simple black hole and solution; and network with cooperative black hole and solution.In both AODV and OLSR-based network, at the begin- ning of simulation, the delay is significant in cases: net- work with simple black hole and solution; and network with cooperative black hole and solution. This delay is the time taken by the source node n0 for checking all received acknowledgments and discovering a new route to reach the destination node nm 1 . Thereafter, graphs become

almost identical and converge to the normal state (without attack). We note that the delay, at the beginning of simulation, in the cooperative black hole case is more significant then that in simple black hole case, because the source node n0 must check all acknowledgments in order to detect the cooperative attack, however, it can detect the simple black hole without checking all acknowl- edgments. Consequently, the cooperative black hole detec- tion takes more time than the simple black hole detection. As a conclusion, the end-to-end delay in our approach con- verge to the normal case without attack.



(a) AODV-based network **(b) OLSR-based network**

Fig. 5. Source traffic sent & destination traffic received.

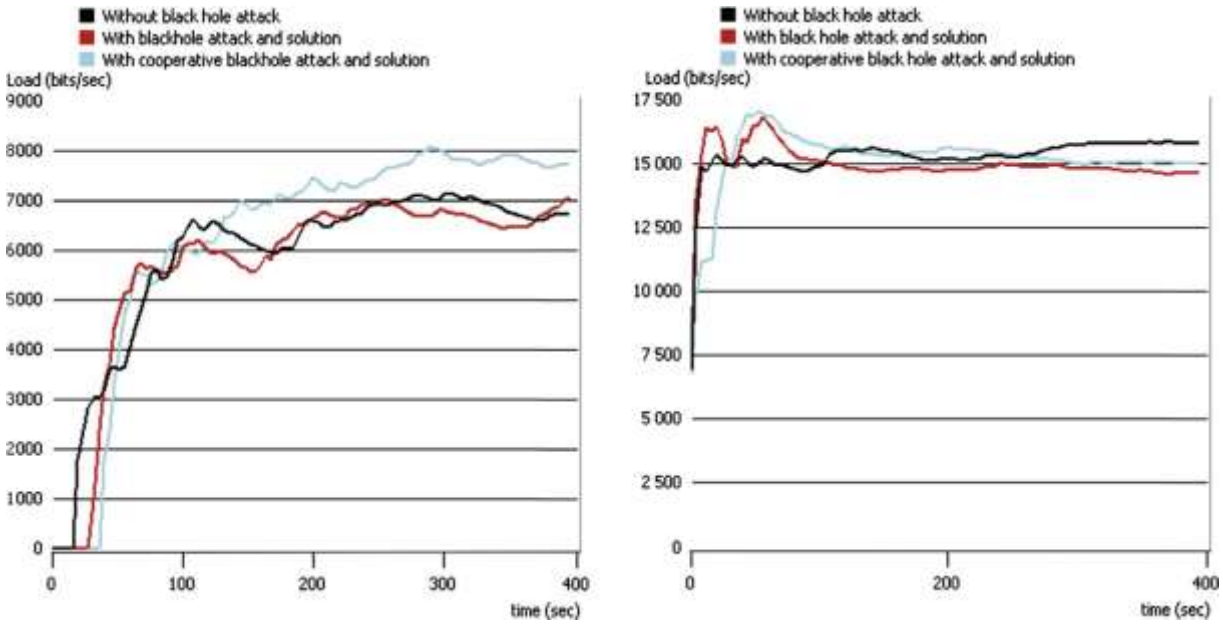


(a) AODV-based network **(b) OLSR-based network**

Fig. 6. End-to-end delay.

Fig. 7 shows the network load in both AODV-based net- work (Fig. 7)) and OLSR-based network (Fig. 7b). We have considered the cases: network without black hole; net- work with simple black hole and solution; and network with cooperative black hole and solution.

In both AODV and OLSR-based network, at the begin- ning of the simulation, the load in the case of a network without attack is small than that in cases of network with attack and solution. This load increase is due to the control information added by our solution, and acknowledgments exchanged among nodes. We note that the load in the cooperative black hole case is more significant than that in the simple black hole case, because, the source node n0 does not need to receive all acknowledgments to detect the simple black hole attack. However, to detect the coop- erative black hole attack, all acknowledgments are needed. Thus, the network load in the cooperative attack case is lar- ger than that in the simple black hole case. Thereafter, the network load in all cases converges to almost the same value. In conclusion, the network load is high at the begin- ning and then it converges to the normal state (case with- out attack).



(a) AODV-based network **(b) OLSR-based network**

Fig. 7. Network load.

VIII. OUR APPROACH VS THE 2-HOP AND THE WATCHDOG APPROACHES

In order to compare our approach to other works, we have chosen the two-hop acknowledgment approach (2- hop ACK) proposed in [3,5,14] and the watchdog [24]. As comparison metrics, we have measured the packets deliv- ery ratio defined as: (number of packets successfully received/number of packets transmitted), the detection ratio and the communication overhead.In Fig. 8, we have plotted the delivery ratio as measured in our approach, the 2-hop ACK and the watchdog. From the figure, we can see that, in both AODV and OLSR-based network, the delivery ratio of our approach is better than that in the 2-hop ACK approach. The reason is that in the 2- hop ACK approach, the detection may be delayed in the case of a selective dropping where the dropper sometimes forwards the packets and sometimes it drops them. This is due to the use of the Bayesian reputation-based approach that gives redemption to suspected nodes as long as they are observed to forward packets. Also, the cooperative black hole attack cannot be detected by the 2-hop ACK approach, which affects

its delivery ratio. The watchdog, which has a multiple drawbacks enumerated in, can- not detect all dropping misbehaviors, therefore its delivery ratio is relatively low.

IX. CONCLUSION

Packets forwarding in multi-hop wireless ad hoc net- work is a cooperative task, in which intermediate nodes participate voluntarily to deliver packets to other nodes. In this paper, we have focused on the black hole attack, where a dishonest node (alone or in cooperation with one or more dishonest nodes) does not forward messages to its successor. The black hole node misbehaves to pre- serve its resources such as its limited energy or to launch a denial of service attack aimed at the network availability. In order to struggle against this attack, we have proposed an authenticated end-to-end acknowledgment based approach, which checks the correct forwarding of packets by intermediate nodes. Our approach detects the black hole launched in simple or cooperative manner, the modification and the replay of messages. Note that the no modification and the no replay of messages are required to fully deliver the message to the destination node. Simulation results shown the detection efficiency and the performance of our approach in both proactive and reac- tive routing protocols based networks. Compared to the 2-hop ACK and the watchdog approaches, our approach has the best delivery ratio of packets and the highest detec- tion ratio, but it generates a communication overhead slightly more significant than that in the 2-hop ACK approach. The approach we propose is a quite resource- demanding, we plan to reduce the generated communica- tion overhead to make it more scalable.

X. REFERENCES

[1] A. Baadache, A. Belmehdi, Avoiding black hole and cooperative black hole attacks in wireless ad hoc networks, Int. J. Comput. Sci. Inform. Security 7 (1) (2010) 10–16.

[2] S. Sharma, R. Gupta, Simulation study of blackhole attack in the mobile ad hoc networks, J. Eng. Sci. Technol. 4 (2) (2009) 243–250.

[3] D. Djenouri, N. Badache, On eliminating packet droppers in MANET:
a modular solution, Ad Hoc Networks 7 (6) (2009) 1243–1258.

[4] P.N. Raj, P.B. Swadas, DPRAODV: a dynamic learning system against blackhole aattack in AODV based MANET, Int. J. Comput. Sci. Issues 2 (2009) 54–59.

[5] D. Djenouri, N. Badache, Struggling against selfishness and black hole attacks in manets, Wireless Commun. Mobile Comput. (WCMC) 8 (6) (2008) 689–704.

[6] H. Weerasinghe, H. Fu, Preventing cooperative black hole attacks in mobile ad hoc networks: simulation, implementation and evaluation, Int. J. Software Eng. Applicat. 2 (3) (2008) 39–54.

[7] L. Tamilselvan, V. Sankaranarayanan, Prevention of cooperative black hole attack in MANET, J. Networks 3 (5) (2008) 13–20.

[8] P. Agrawal, R.K. Ghosh, S.K. Das, Cooperative black and gray hole attacks in mobile ad hoc networks, in: Proceedings of the 2nd International Conference on Ubiquitous Information Management and Communication (ICUIMC'08), 2008, pp. 310–314.