MODSBR: An Enhanced Methodology for Defending Byzantine Attacks on Mobile Ad Hoc Networks

Safaa Saad-Eldeen Ahmed
Faculty of computers and Information, Information Technology Department
safee_2000@hotmail.com

Wail Shawky Elkilani
Faculty of computers and Information, Information Technology Department
welkilani@gawab.com

Mohiy Mohammed Hadhoud
Faculty of computers and Information, Information Technology Department
mmhadhoud@yahoo.com

Abstract: *Byzantine attacks signify a risk to Mobile Ad Hoc Networks (MANETs). One of the efficient routing protocols used to mitigate such attacks in MANETs is the On-Demand Secure Byzantine Routing protocol (ODSBR). In this paper, we present the Modified ODSBR (MODSBR) where several enhancements for the performance and security of ODSBR are implemented. The first of these enhancements is enabling route caching in an elegant way that causes a notable decrease in the routing overhead. The central clustered link weight management (CLWM) is a secure technique utilized by MODSBR to detect malicious nodes through a set of management nodes (MN). The effect of the proposed enhancements is evaluated with respect to different types of Byzantine attacks. Simulation experiments show that MODSBR outperforms ODSBR by an average 23% decrease in routing overhead and a 10 % increase in the delivery ratio for different mobility values.*

Keywords: Secure routing; Caching; Mobile Ad Hoc Networks.

1. Introduction

In ad hoc networks, devices rely on each other to keep the network connected. Thus, unlike traditional wireless solutions, such networks do not require any pre-existent (fixed) infrastructure, which minimize their cost and deployment time. Routing protocols enable multi-hop communications in ad hoc networks. To achieve availability, routing protocols should be robust against both topology changes and malicious attacks.

Existing protocol specifications cope well with the change of network topologies. However, defense against malicious attacks has remained optional. Nowadays, the trend is changing and there is an increasing interest on research focused on the provision of proposals for securing ad hoc routing protocols.

Attacks where the adversary has full control of an authenticated device and can perform arbitrary behavior to disrupt the system is called, Byzantine attacks. From a more general perspective, a Byzantine attack is any attack that involves the leaking of authentication secrets so that an adversarial device is indistinguishable from a legitimate one. Significant research in securing wired [7, 4, 13] or ad hoc wireless [10, 9, 17, 16] routing protocols focused on this aspect. In this work, only Byzantine attacks were considered. It is believed that they represent type of attacks that are likely to be mounted against ad hoc wireless routing protocols. And they cover a wide range of adversarial strengths. Individual techniques are proposed [14, 11, 12, 1, 10] to mitigate each type of these attacks. A few research efforts are done for a global prevention solution.

Black hole Attack: where the adversary stops forwarding data packets, but still participates in the routing protocol correctly. As a result, whenever the adversarial node is selected as part of a path by the routing protocol, it prevents communication on that path from taking place. Several techniques exist which attempt to mitigate the effect of black hole attacks on network performance. One of these methods is Watchdog and Pathrater [14]. The approach has two components, watchdog, a service that is run by each node and monitors the node's neighbors, and pathrater, a service that ensures that adversarial nodes are avoided when selecting future routes. An alternate method for avoiding black hole attacks is the Secure Data Transmission (SDT) protocol [15]. SDT uses authenticated end-to-end acknowledgments from the final destination, providing proof that the packets reached their destination.

Flood Rushing Attack: exploits the flood duplicate suppression technique used by many routing protocols. This attack takes place during the propagation of a legitimate flood and can be seen as a "race" between the legitimate flood and the adversarial variant of it. If an adversary successfully reaches some of its neighbors with its own version of the flood packet before they receive a version through a legitimate route, then those nodes will ignore the legitimate version and will propagate the adversarial version. This may result in the continual inability to establish an adversarial-free route, even when authentication techniques are used. Previous work in addressing the rushing attack is scarce, we are only aware of Rushing Attack Prevention (RAP) [12]. The intuition in this work is that the rushing attack can be prevented by waiting (up to a time limit □) to receive up to □□requests (flood re-broadcasts) and then randomly selecting one to forward rather than always forwarding only the first one.

Byzantine Wormhole Attack: where two adversaries collude by tunneling packets between each other in order to create a shortcut (or wormhole) in the network. This tunnel can be created either using a private communication channel, such as a pair of radios and directional antennas, or by using the existing ad hoc network infrastructure. The adversaries can send a route request and discover a route across the ad hoc network, then tunnel packets through the non-adversarial nodes to execute the attack. . A mechanism called Packet Leashes for preventing wormholes by limiting the transmission distance of a link is proposed in [11]. And also Directional Antenna is a more recent method for preventing wormholes by using the angle of arrival information available when using directional antennas [8].

Byzantine Overlay Network Wormhole Attack: A more general variant of the previous attack occurs when several nodes are compromised and form an overlay network. By tunneling packets through the overlay network, the adversaries make it appear to the routing protocol that they are all neighbors, which considerably increases their chances of being selected on routes. The same prevention techniques of preventing the wormhole attack are utilized.

ODSBR [1-3] routing protocol is a very effective secure on-demand routing protocol that is resilient to Byzantine failures caused by individuals or colluding nodes. An adaptive probing technique is used that detects a malicious link after log n faults occurred, where n is the length of the path. These links are then avoided by multiplicatively increasing their weights and by using an on-demand route discovery protocol that finds a least weight path to the destination. Disabling the route caching property is a factor that causes decrease in the performance of the ODSBR protocol. We believe that techniques taking advantage of route caching will enhance the performance. Also, one of the disadvantages of this protocol is that when any node detects a link that caused failure, it doesn't tell other nodes about this unreliable link. So a technique taking care of this aspect would increase the security. Implementation details are also discussed, as well as changes to the original protocol motivated by practical considerations. The rest of the paper is organized as follows. Section 2 describes the caching technique and its implementation. The central clustered link weight management technique is presented in section 3. Section 4 presents analysis of the simulation results. We conclude the work in Section 5.

2. Applying Route Caching Mechanism to ODSBR Routing Protocol

Due to power limitation each station has a fixed range. It also acts as a router, relaying data packets for other stations to their final destination. One of the main challenges in the design of ad hoc networks is the routing protocol upon a dynamically changing topology, node energy constraints and the properties of the wireless channel. On-demand routing protocol that is generally used in ad hoc networks is one that searches for a route to a destination node when a sending node originates a data packet addressed to the destination node. Every on-demand routing protocol has to maintain some form of routing cache with the intention of avoiding route re-discoveries for each separate data packet and reduction of routing overhead. Additionally, the route cache is not only used to cache routes for the purpose of originating packets, but also for the purpose of allowing nodes to answer Route Requests targeted at other node. Therefore, caching is an essential component of on-demand routing protocol for wireless ad hoc mobile networks.

ODSBR routing protocol is a very effective secure on-demand routing protocol that is resilient to Byzantine failures. But disabling the route caching property is a factor that causes degradation in the performance of the ODSBR protocol. In order to enhance the performance, we investigated ways of taking advantage of route caching.

In the original ODSBR protocol, as shown in Figure 1, when a node receives a route_request it will check if this is a new request or repeated one. If the request is repeated, it will be discarded. Otherwise the signature will be verified. The node will broadcast a route reply for verified signatures if the requested destination ID is the same ID of this node, otherwise the route request will be broadcasted.

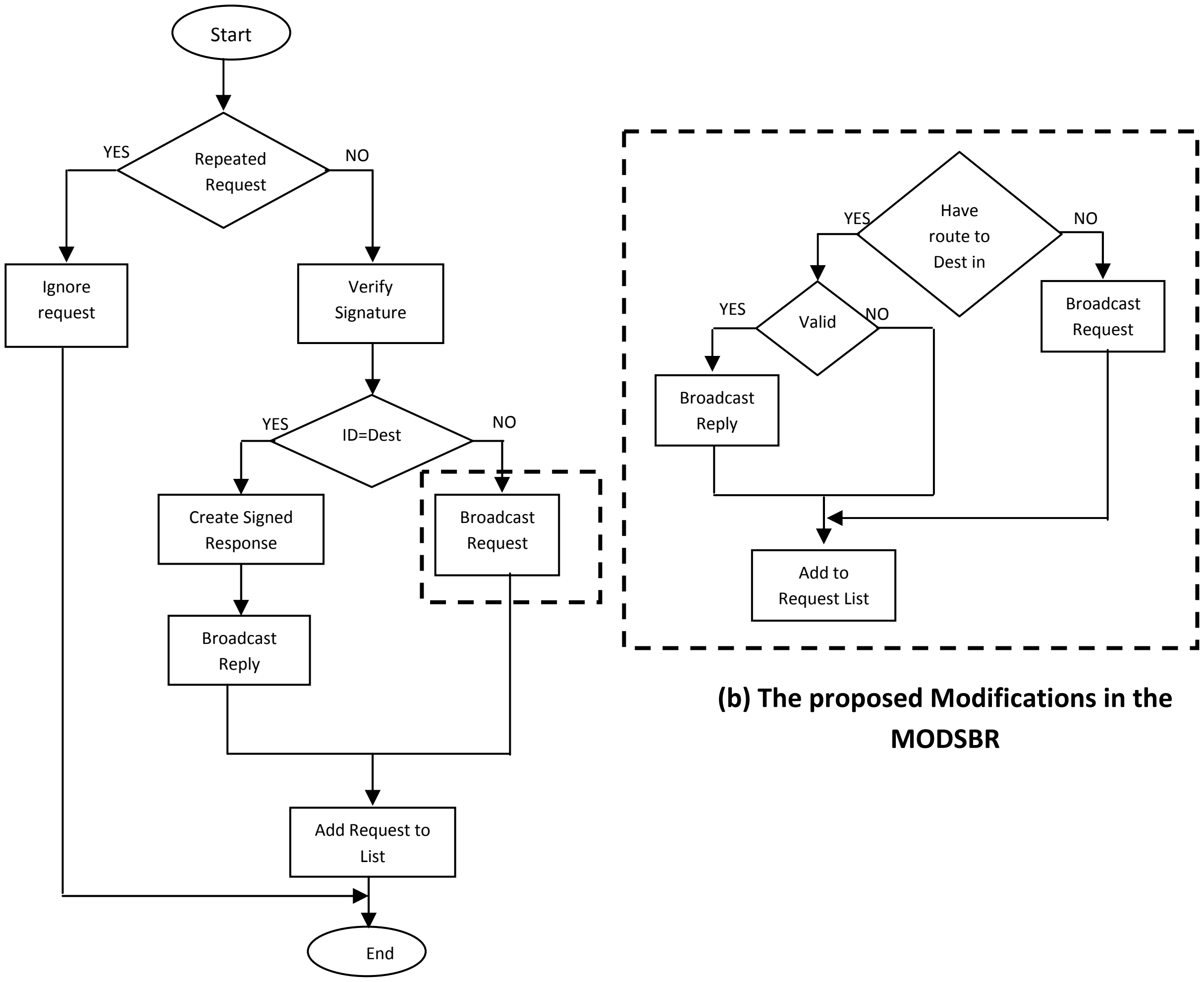
After applying the caching technique the ODSBR will behave as follows: when a node receives a route_request it will check if this is new request or repeated. If the request is repeated, it will be discarded. Otherwise the signature will be verified. Then if the requested destination ID is the same ID of this node, request will be broadcasted for absent routes in the cache. On the other hand, the present route will be validated. Reply will be broadcasted for valid routes.

3. Central Clustered Link Weight Management

In the original ODSBR, every node has its own link weight table. When any node wants to select the best route to a specific destination after a route discovery operation, it will compute the total weight for each path received in a route reply message and then select the path with the smallest weight value. When the value of the loss rate at any node becomes greater than 10%, the node will start the binary search to detect the link that caused loss in packets. When the node finds the lossy link, it will increment the weight for this link in the weight table specialized for this node.

On the other hand, instead of making each node to have a link weight table and when any table has been updated, no way to any node to know about this update. So, a set of nodes called Management nodes (MN) had been specified to store a general weight table, where any node can use this table during computing the total weight to any path. Also, when any node detects a link that caused loss of data, it will increment the weight value of this link in the nearest general weight table. And so on, each node makes updates for links` weights in the general link weight table and also uses this table for selecting the best path.

Normally the number of management nodes is significantly less than the total number of nodes (n). We have found that number of MN giving best performance will be about 20% of the nodes. They are basically normal nodes except that they carry the general management tables and able to be accessed by more than one node in the same time.



(b) The proposed Modifications in the MODSBR

(a) ODSBR Behavior when node receives a request

Figure 1. The Route Request Behavior

We have slowed down the movement of the management nodes such that the same set of nodes are served during simulation. Figure 2 shows the algorithm of the central clustered link weight management technique.

It's assumed that every node has an attribute defined as management flag. For nodes that work as management nodes this flag is set to “1” and nodes that work as non management nodes is set to “0”. Each node configures this flag offline before entering the network. Also it's not allowed that all nodes work as non Management nodes, there must be some of nodes to work as Management nodes.

```
Initialize all weight of nodes with zero in all Management nodes
-When any node wants to compute the total weight for a specific path
  •   Define list as an array.
  •   Add all nodes in the path to list.
  •   Request the weight of the nodes involved in the list by sending weight_request packet attached
      with the list array to the nearest MN.
-When any node detects a malicious node that caused failure
  • Increment the weight value of this node causing failure.
  • Send an update_packet attached with the link new weight value to the nearest Management node,
    notice that nearest MN is detected by hello packets.
-When any MN receives an update_packet
  • Clear the old weight of this link.
  • Set the weight of this link to the new weight.
-When any MN receives a weight request for nodes in the list array
  • Receive the list of links.
  • Define weight_list as an array.
  • For each item in list array
       - Search for the weight value.
       - Add this weight to the weight_list array.
  • Send a weight_reply packet attached with the weight_list array.

  Notice: - that detection will be fulfilled as implemented in the ODSBR
```

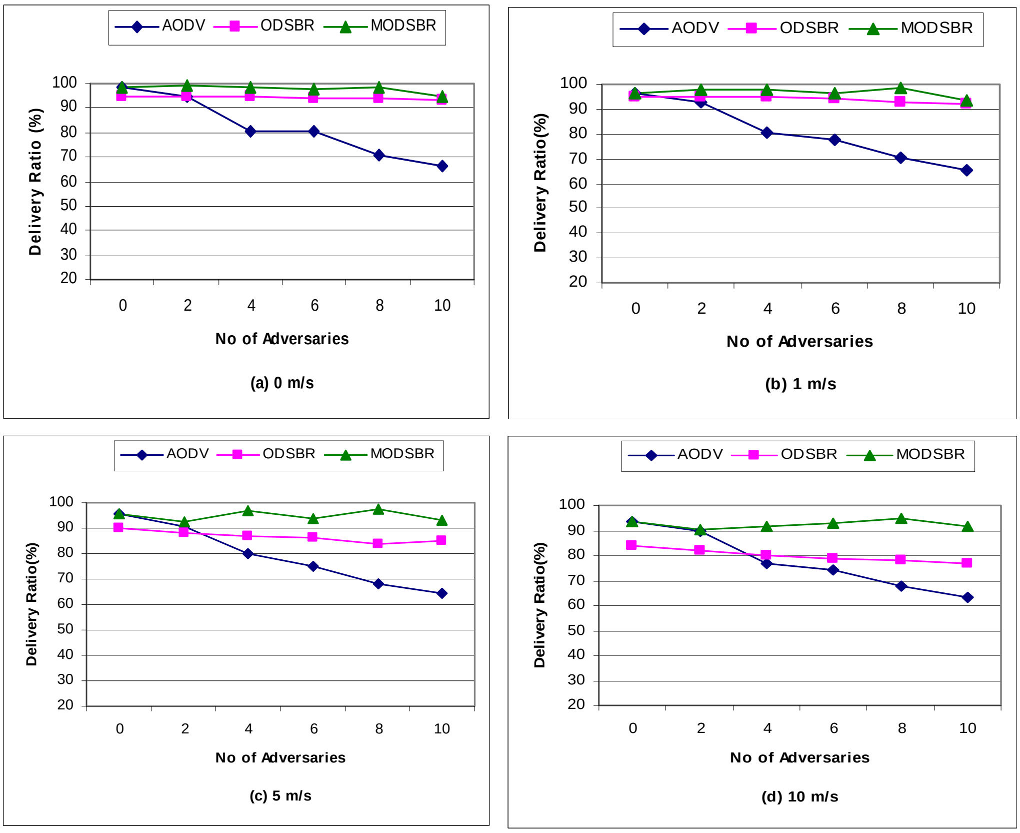
Figure 2. An algorithm for central link weight management technique

4. Results

Simulations were conducted using the NS2 [18] network simulator. Nodes in the network were configured to use 802.11 radios with a bandwidth of 2 Mbps and a nominal range of 250 m. All the simulated routing protocols were configured with their default parameters. The simulations were conducted by randomly placing 50 nodes within a 1000 by 1000 meter square area. In addition to these 50 nodes, 0 to 10 adversarial nodes were added to the simulations, depending on the considered attack configuration. A traffic load of 10 constant bit rate (CBR) flows was used to simulate data communication through the ad hoc network. An aggregate load of 0.1 Mbps was offered to the network by having each flow send 256 byte packets at approximately 4.9 packets per second. The simulation time was 300 seconds for each simulation and the results were averaged over 30 random seeds. The next subsections will show the result of applying the MODSBR to defend different Byzantine attacks.

5. The Byzantine Black hole Attack

The delivery ratio was evaluated by using as a baseline the case where no black holes exist in the network. The number of adversarial nodes was then increased in the network and the effect on the delivery ratio was evaluated. The adversarial nodes were placed randomly within the simulation area. Figure 3 shows the delivery ratio of the AODV [5, 6], the ODSBR, and modified ODSBR protocols as a function of the number of adversarial nodes, for different levels of mobility. It can be noticed that at 0 m/s and 1 m/s speeds the MODSBR outperform ODSBR and AODV. Also the MODSBR increased the delivery ratio over 90% at high mobility.



(a) 0 m/s

(b) 1 m/s

(c) 5 m/s

(d) 10 m/s

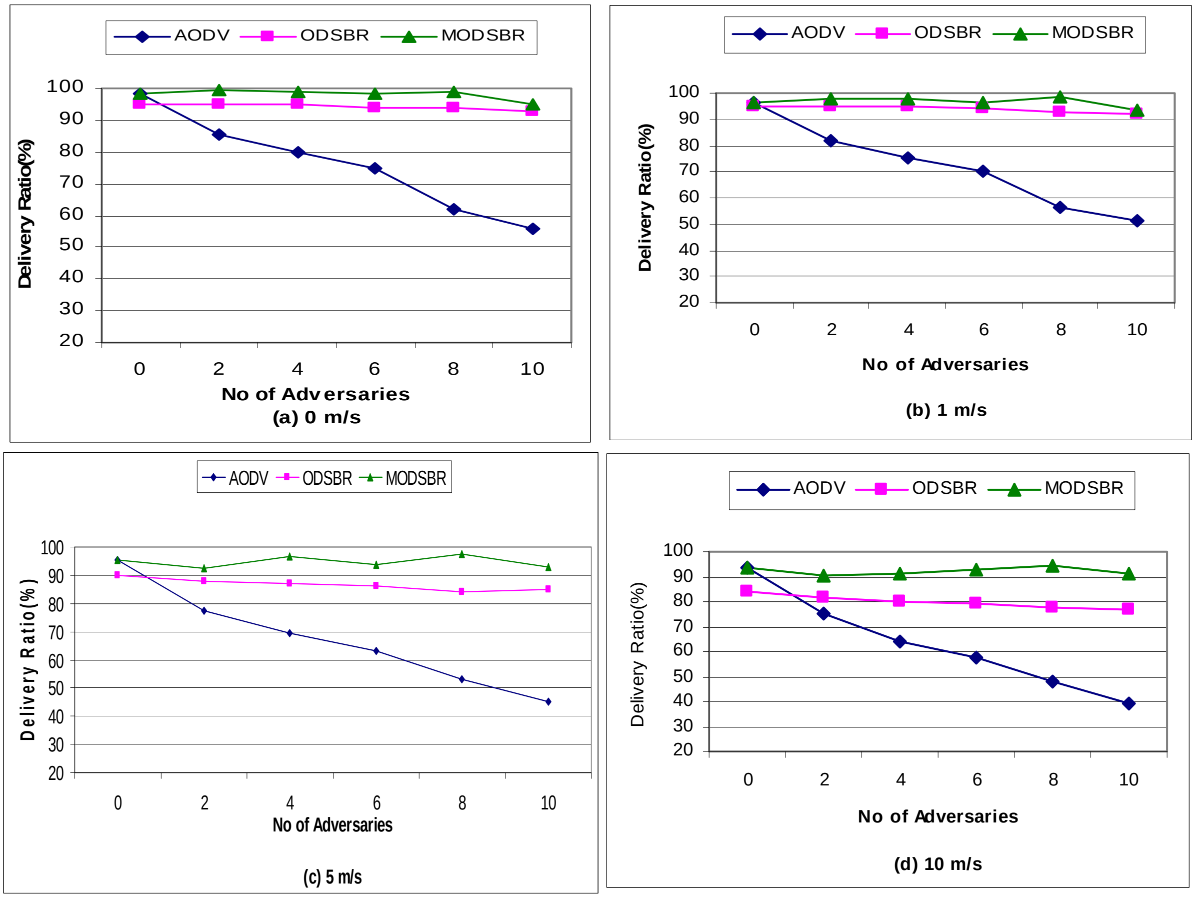
Figure 3. The effect of the Black Hole Attack on the delivery ratio of AODV, ODSBR, and MODSBR for different speeds

To compare between the performance of the MODSBR and ODSBR, we have used % delivery ratio error defined as %DRE = (DRn – DRo) / DRo where DRn is the delivery ratio for the ODSBR or the MODSBR. And DRo is the delivery ratio for the AODV. Table 1 compares between the ODSBR & MODSBR using the %DRE after applying the black hole attack. It can be noticed that ODSBR couldn't improve delivery ratio in some cases due to high mobility while the MODSBR improved the delivery ratio more than the ODSBR for high and low mobility.

	ODSBR				**MODSBR**			
No of adversaries	**0 m/s**	**1 m/s**	**5 m/s**	**10 m/s**	**0 m/s**	**1 m/s**	**5 m/s**	**10 m/s**
2	0.00	0.03	-0.03	-0.08	0.05	0.05	0.03	0.01
4	0.18	0.18	0.09	0.04	0.23	0.22	0.21	0.19
6	0.16	0.21	0.15	0.07	0.21	0.24	0.25	0.26
8	0.32	0.32	0.23	0.15	0.39	0.40	0.43	0.39
10	0.40	0.40	0.32	0.22	0.43	0.42	0.44	0.45

Table 1. %DRE for ODSBR & MODSBR in the presence of the Black Hole Attack

Simulations were conducted to evaluate the impact of flood rushing on the effectiveness of a black hole attack. Figure 4 shows the delivery ratio of the AODV, ODSBR and the MODSBR protocols as a function of the number of adversarial nodes, for different mobility values in the presence of both the black hole attack and the flood rushing attack. As shown in the figures the impact of flood rushing on MODSBR is almost unnoticeable.



(a) 0 m/s

(b) 1 m/s

(c) 5 m/s

(d) 10 m/s

Figure 4. The effect of the Black Hole Attack combined with flood rushing Attack on the delivery ratio of AODV, ODSBR, and MODSBR for different speeds

Table 2 compares between the ODSBR & MODSBR using the %DRE after applying the black hole attack in combination with flood rushing attack. We can notice that in spite of increasing the no of adversaries the MODSBR could improve the delivery ratio.

	ODSBR				MODSBR			
No of adversaries	0 m/s	1 m/s	5 m/s	10 m/s	0 m/s	1 m/s	5 m/s	10 m/s
2	0.11	0.16	0.14	0.09	0.16	0.20	0.20	0.20
4	0.19	0.27	0.25	0.24	0.24	0.30	0.39	0.42
6	0.26	0.34	0.36	0.37	0.31	0.38	0.48	0.61
8	0.52	0.64	0.58	0.63	0.59	0.74	0.84	0.97
10	0.66	0.80	0.89	0.95	0.70	0.83	1.06	1.32

Table 2. %DRE for ODSBR & MODSBR in the presence of the Black Hole Attack & Flood Rushing.

5.1. Byzantine Wormhole Attacks

The wormhole attack can be implemented by three forms central wormhole, cross of death wormhole, and the random placement attack. In the case of central wormhole configuration only two adversaries placed at coordinates (300,500) and (700,500) in the 1000 x 1000 simulation area as shown in figure 5. In case of cross of death configuration four adversaries are placed at coordinates (200,500), (800,500), (500,200), (500,800). They form two wormholes, in the shape of a cross (see figure 6). In the last configuration a set of wormholes is randomly placed in the network. In all cases, we have first evaluated the effect of the wormhole attack on the delivery ratio. We then combined the wormhole with flood rushing and examined the impact of the combined attack.

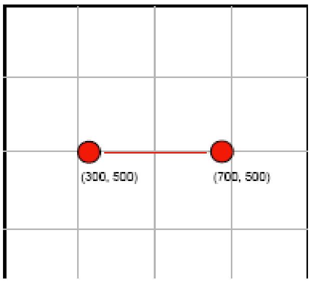


Figure 5. Central Wormhole

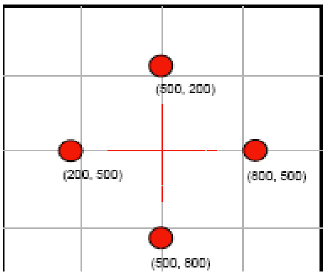
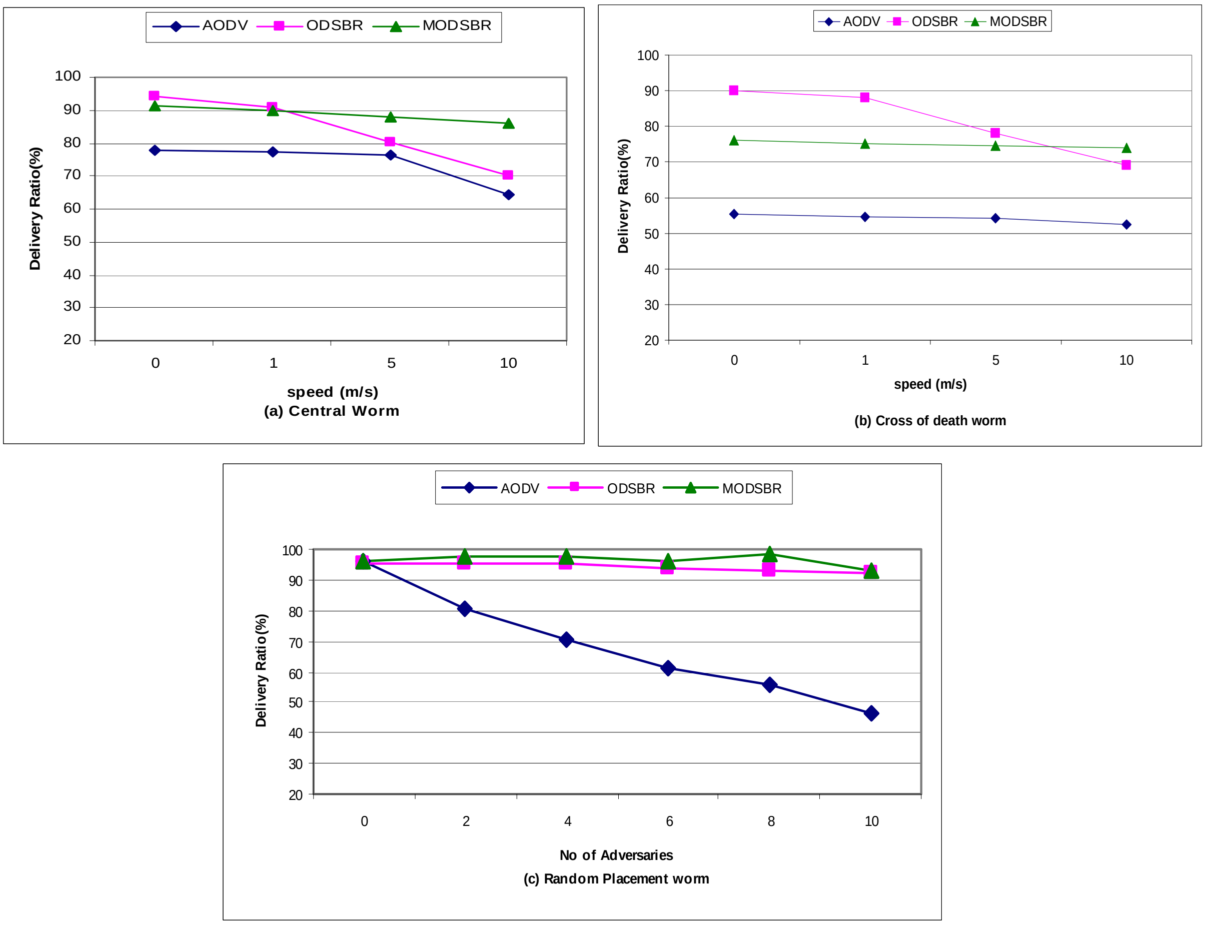


Figure 6. Cross of Death

Figure 7 shows the Delivery ratio for AODV, ODSBR, and MODSBR after applying each form of the wormhole attack. As noticed in case of central wormhole (Figure 7.a) the delivery ratio decreased in low mobility in case of the MODSBR more than the ODSBR. As shown in Figure 7.a that for low mobility the delivery ratio in case of the MODSBR is less than the ODSBR. For high mobility, the DR of the ODSBR has decreased about 20%, while that of the MODSR is nearly constant and stable. The same discussion can be drawn for the cross of death attack as shown in Figure 7.b where the constant value of the MODSBR can be noticed while deterioration in the performance of the ODSBR can be observed. This can be explained by that because the node still under the effect of the attackers in low mobility and the attacker may prevent it from connecting to the management nodes around it. But in high mobility it can leave the area of attacker. For the random placement wormhole (Figure 7.c), the MODSBR performs slightly better than the ODSBR.



(a) Central Worm

(b) Cross of death worm

(c) Random Placement worm

Figure 7. The effect of the Wormhole Attack in different forms on the delivery ratio of AODV, ODSBR, and MODSBR

Table 3 compares between the ODSBR & MODSBR using the %DRE for the central wormhole and cross of death wormhole. In case of the central wormhole, we can notice that MODSBR could improve the delivery ratio for high mobility while the ODSBR performs better in low mobility. Also in the case of cross of death it's clear that ODSBR give good delivery ratio compared to the MODSBR.

	ODSBR		MODSBR	
Speed	Center	Cross of Death	Center	Cross of Death
0 m/s	0.21	0.63	0.17	0.38
1 m/s	0.18	0.61	0.16	0.38
5 m/s	0.05	0.44	0.15	0.38
10 m/s	0.09	0.32	0.34	0.41

Table 3. %DRE for ODSBR & MODSBR in the presence of the central wormhole & the cross of death worm

Table 4 compares between the ODSBR & MODSBR using the %DRE for the random placement wormhole. It's clear that MODSBR improved the delivery ratio values more than the ODSBR with any no of adversaries for allmobility levels.

	ODSBR				MODSBR			
	0 m/s	1 m/s	5 m/s	10 m/s	0 m/s	1 m/s	5 m/s	10 m/s
2	0.16	0.18	0.14	0.15	0.21	0.21	0.20	0.27
4	0.34	0.35	0.25	0.21	0.40	0.394	0.39	0.38
6	0.51	0.54	0.42	0.32	0.58	0.58	0.55	0.56
8	0.70	0.68	0.52	0.48	0.78	0.78	0.76	0.80
10	0.91	0.97	0.85	0.67	0.95	1.00	1.02	0.99

Table 4. %DRE for ODSBR & MODSBR in the presence of the random placement wormhole

Figure 8 shows the delivery ratio of the AODV, ODSBR, and MODSBR after applying the wormhole attack combined with the flood rushing attack. It can be noticed that the performance of the MODSBR in the presence of flood rushing gets worse. In fact it gives a worse response compared to the ODSBR.

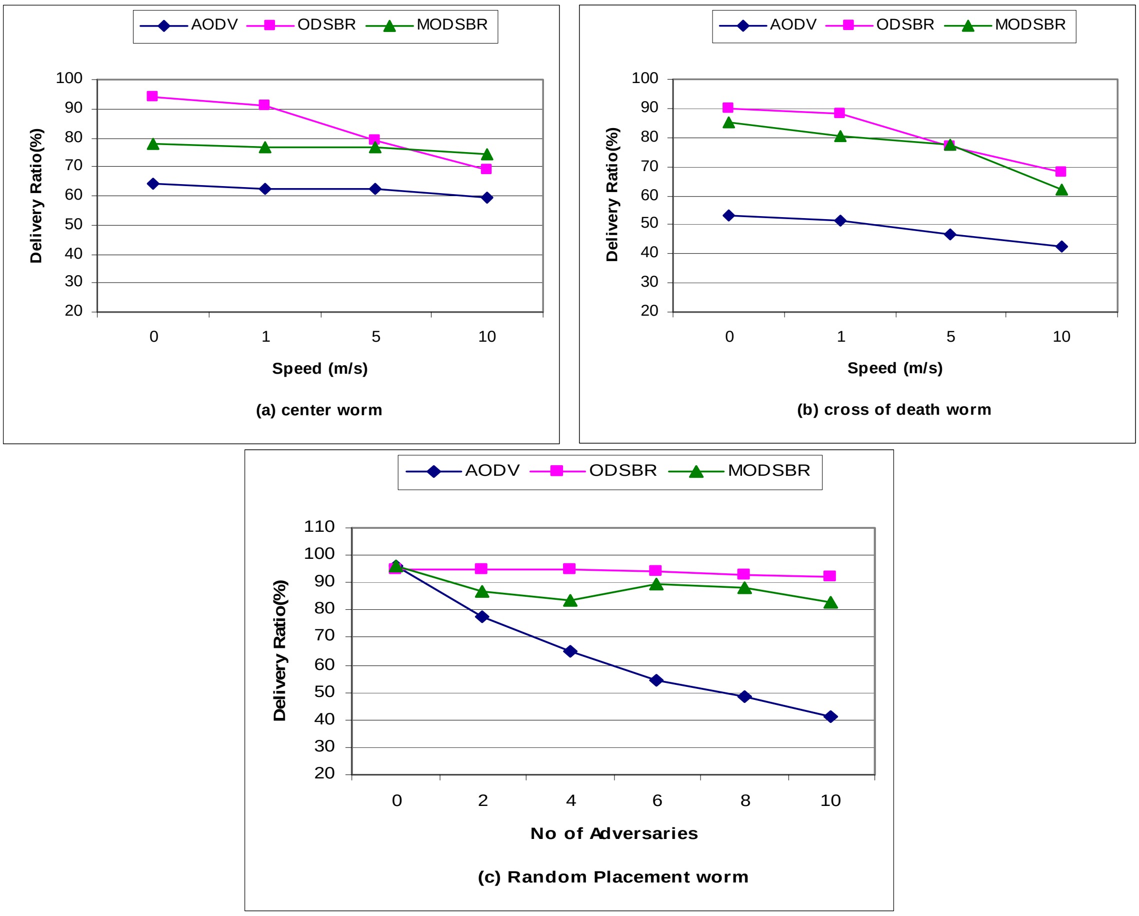


Figure 8. The effect of the Wormhole Attack in different forms combined with the flood rushing on the delivery ratio of AODV, ODSBR, and MODSBR

Table 5 compares between the ODSBR & MODSBR using the %DRE with the presence of central wormhole or cross of death wormhole in combination with the flood rushing. As shown the MODSBR doesn't perform well with the presence of the central wormhole & flood rushing but the ODSBR improves the delivery ratio more in this case. In the case of cross of death with flood rushing the ODSBR and MODSBR have values close to each other.

	ODSBR		**MODSBR**	
	Center	**Cross of Death**	**Center**	**Cross of Death**
0 m/s	0.47	0.69	0.22	0.60
1 m/s	0.45	0.71	0.23	0.56
5 m/s	0.27	0.65	0.23	0.66
10 m/s	0.16	0.59	0.25	0.46

Table 5. %DRE for ODSBR & MODSBR in the presence of each central wormhole & cross of death worm in combination with Flood Rushing

Table 6 compares between the ODSBR & MODSBR using the %DRE with the presence of random placement wormhole in combination with the flood rushing. As shown the MODSBR improved the delivery ratio values in a noticeable way but when compared to the ODSBR, the ODSBR is a little better.

	ODSBR				**MODSBR**			
	0 m/s	**1 m/s**	**5 m/s**	**10 m/s**	**0 m/s**	**1 m/s**	**5 m/s**	**10 m/s**
2	0.21	0.23	0.19	0.20	0.14	0.12	0.16	0.19
4	0.45	0.47	0.36	0.31	0.38	0.29	0.30	0.34
6	0.70	0.72	0.59	0.48	0.66	0.63	0.65	0.55
8	0.93	0.91	0.73	0.70	0.84	0.80	0.73	0.82
10	1.26	1.22	1.12	0.93	1.04	0.99	1.03	0.98

Table 6. %DRE for ODSBR & MODSBR in the presence of the random placement wormhole & flood rushing

5.2. Byzantine Overlay Network Wormhole Attacks

In this section, we will evaluate the damage caused to AODV by a set of attackers performing a coordinated super-wormhole attack, and demonstrate the effectiveness of the MODSBR and ODSBR protocol in mitigating this attack. Similar to the wormhole attack we investigated three configurations namely the cross of death, random placement and complete coverage. The same configurations of the cross of death and random placement were applied. In the complete coverage the adversaries attempt to arrange themselves so that their combined communication areas completely cover the full ad hoc network. This means that if any transmission takes place in the network, an adversary will hear it. We simulated the configuration shown in Figure 9, with five adversarial nodes placed at coordinates (250,250), (250,750), (500,500), (750,250), (750,750).

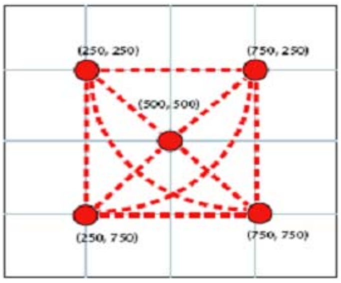


Figure 9 Complete Coverage Configuration

Figure 10 shows the delivery ratio of the AODV, ODSBR, and MODSBR after applying the three configurations of the overlay wormhole attack. In the case of cross of death and complete coverage it can be noticed that the delivery ratio of the ODSBR is better than the delivery ratio of the MODSBR except for high mobility. On the other hand, the delivery ratio of the ODSBR had decreased as the mobility increased, while for high mobility the MODSBR is approximately constant as with low mobility. In the case of random placement the delivery ratio has increased over than 90% even if the mobility increased.

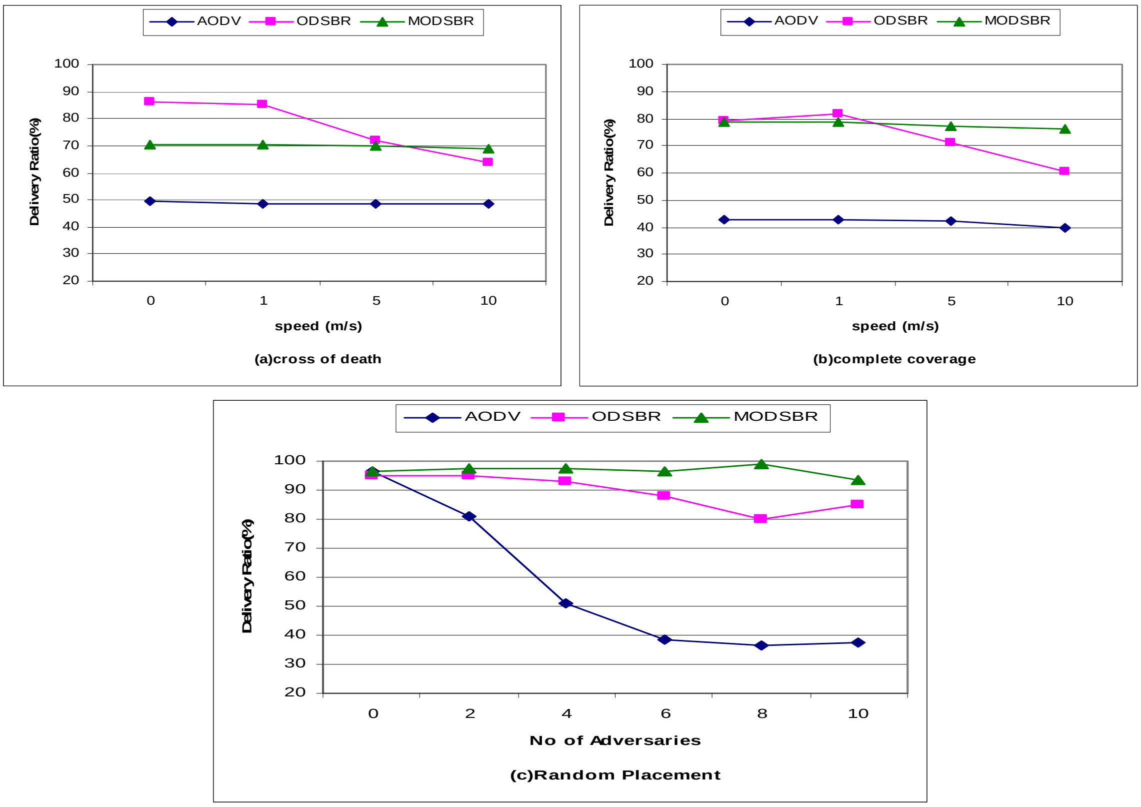


Figure 10. The effect of the overlay wormhole attack in different forms on the delivery ratio of AODV, ODSBR, and MODSBR

Table 7 compares between the ODSBR & MODSBR using the %DRE in the presence of complete coverage or cross of death. We can notice that MODSBR has improved the delivery ratio only for high mobility in these two cases.

	ODSBR		MODSBR	
Speed	**Cross of Death**	**Complete Coverage**	**Cross of Death**	**Complete Coverage**
0 m/s	0.73	0.84	0.42	0.84
1 m/s	0.75	0.92	0.44	0.85
5 m/s	0.48	0.68	0.44	0.82
10 m/s	0.32	0.52	0.43	0.91

Table 7. %DRE for ODSBR & MODSBR in the presence of each complete coverage & cross of death overlay wormhole

Table 8 compares between the ODSBR & MODSBR using the %DRE in the presence of random placement overlay wormhole. We can notice that MODSBR has improved the delivery ratio even for high and low mobility with respect to all no of adversaries.

	ODSBR				MODSBR			
No of adversaries	**0 m/s**	**1 m/s**	**5 m/s**	**10 m/s**	**0 m/s**	**1 m/s**	**5 m/s**	**10 m/s**
2	0.14	0.17	0.19	0.09	0.19	0.20	0.25	0.24
4	0.79	0.82	0.63	0.53	0.91	0.91	0.89	0.86
6	1.27	1.30	1.15	0.99	1.47	1.52	1.55	1.57
8	1.20	1.18	1.32	1.24	1.47	1.70	2.10	2.07
10	1.15	1.26	0.94	0.78	1.49	1.48	1.50	1.50

Table 8. %DRE for ODSBR & MODSBR in the presence of random placement overlay wormhole

Figure 11 shows the delivery ratio of the AODV, ODSBR, and MODSBR after applying each configuration of the overlay wormhole attack combined with the flood rushing attack. In the case of cross of death and complete coverage it can be noticed that the delivery ratio of the ODSBR is better than the delivery ratio of the MODSBR except for high mobility. On the other hand, the delivery ratio of the ODSBR has decreased as the mobility increased, but the MODSBR gives approximately the same DR for high mobility. In the case of random placement the delivery ratio has increased over than 90% even if the mobility increased. It can be noticed that performance for this case is similar to the overlay wormhole results.

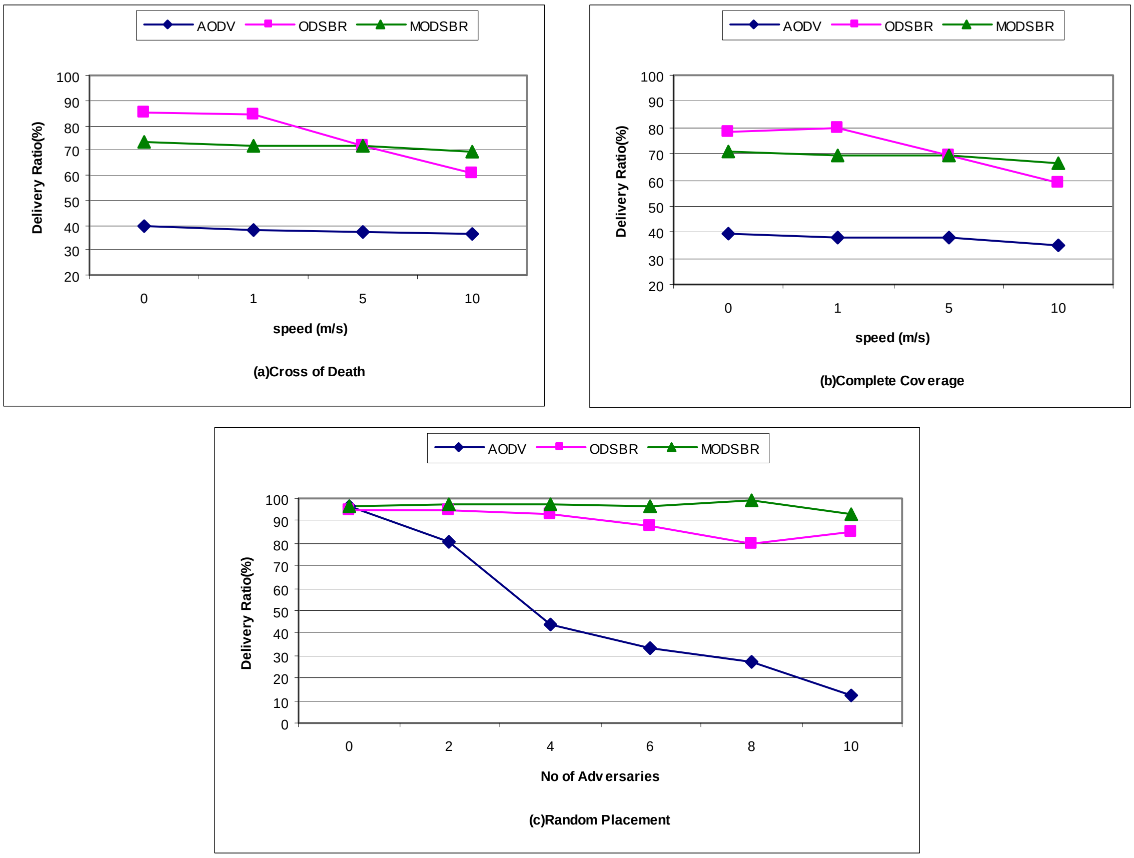


Figure 11. The effect of the overlay Wormhole Attack in different forms combined with the flood rushing on the delivery ratio of AODV, ODSBR, and MODSBR

Table 9 compares between the ODSBR and MODSBR using the %DRE when applying each of the complete coverage overlay wormhole and cross of death wormhole combined with flood rushing. Also here the MODSBR performs well for high mobility but in low mobility the ODSBR is better.

	ODSBR		MODSBR	
	Cross of Death	Complete Coverage	Cross of Death	Complete Coverage
0 m/s	0.73	0.84	0.87	0.80
1 m/s	0.75	0.92	0.87	0.82
5 m/s	0.48	0.68	0.92	0.82
10 m/s	0.32	0.52	0.88	0.87

Table 9. %DRE for ODSBR & MODSBR in the presence of each complete coverage & cross of death overlay wormhole with flood rushing

Table 10 compares between the ODSBR and MODSBR using the %DRE when applying the random placement overlay wormhole in combination with flood rushing. It can be noticed that the MODSBR improved the delivery ratio values more than the ODSBR in spite of presence of the flood rushing attack.

	ODSBR				MODSBR			
	0 m/s	1 m/s	5 m/s	10 m/s	0 m/s	1 m/s	5 m/s	10 m/s
2	0.16	0.17	0.27	0.18	0.22	0.20	0.33	0.33
4	0.81	1.12	1.10	1.12	0.92	1.23	1.44	1.59
6	1.62	1.66	1.41	2.04	1.85	1.91	1.86	2.93
8	2.16	1.91	1.82	2.33	2.56	2.59	2.77	3.57
10	5.00	6.08	8.11	7.71	5.96	6.78	10.75	11.25

Table 10. %DRE for ODSBR & MODSBR in the presence of random placement overlay wormhole & flood rushing

To conclude the previous results of all attacks, we can see the average of the delivery ratio values in Table 11. From the values we can say that the MODSBR has improved the delivery ratio more than the ODSBR by approximately 10% on the average. To be noted %average delivery ratio is calculated according to [(DRMODSBR - DRODSBR) / DRODSBR].

Attacks	**ODSBR**	**MODSBR**	**% average delivery ratio**
Black hole	86.75	95.4951	% 6.745102
Black hole Rushing	87.75	95.4951	% 6.7451
Wormhole center	83.75	89.7791	% 5.029147
Wormhole random	90.666	95.495	% 3.828435
Overlay Worm Complete Coverage	72.125	78.77	% 4.652532
Wormhole random	81.654	96.4951	% 11.82843
Wormhole random rush	82.666	96.6849	**% 13.01827**
Averages	86.75	95.4955	**% 9.6**

Table 11. Average delivery ratio for all experimented attacks

5.3. Routing Overhead

Simulations were conducted to compare the overhead of ODSBR with that of AODV, in order to evaluate the cost of security. In addition to route discovery overhead before and after applying route caching to ODSBR, ODSBR requires a protocol acknowledgment for each successfully delivered data packet. In real implementations, ODSBR acknowledgments can be piggy-backed on TCP acknowledgment packets, thus we only consider routing packets in the overhead measurements.

5.3.1. Simulation Results

Figure 12 illustrates the overhead of ODSBR and MODSBR in a non-adversarial scenario at all levels of mobility.

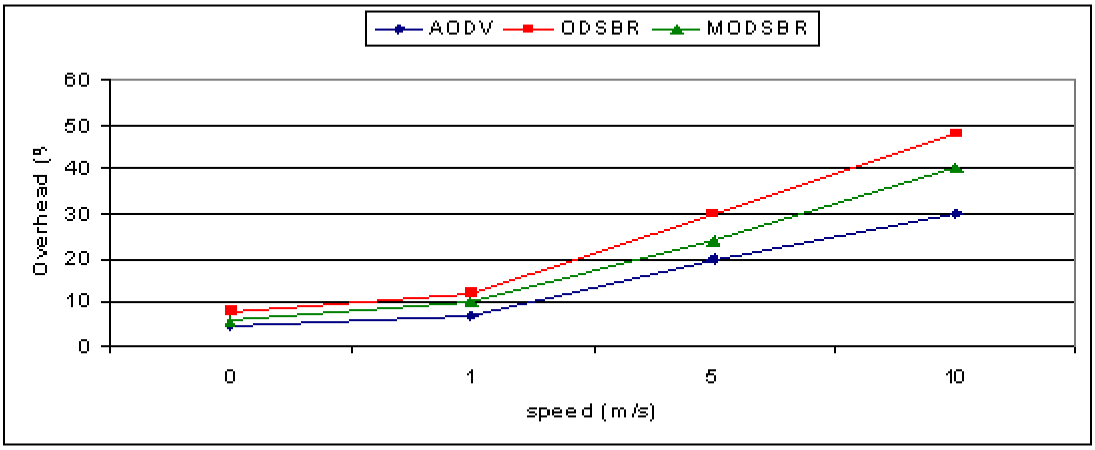


Figure 12. The overhead for the ODSBR and MODSBR without any attacks

Figure 13 depicts the overhead of the routing protocols as a function of the number of adversaries, when the adversaries execute a black hole attack. The nodes are under random way-point mobility with a maximum speed of 1 m/s. Observe that the routing overhead of MODSBR increases with the number of adversaries. This occurs as a result of the protocol activity in detecting faults and readjusting the path to avoid them. The overhead of MODSBR increases proportionally to the number of faulty links in the network.

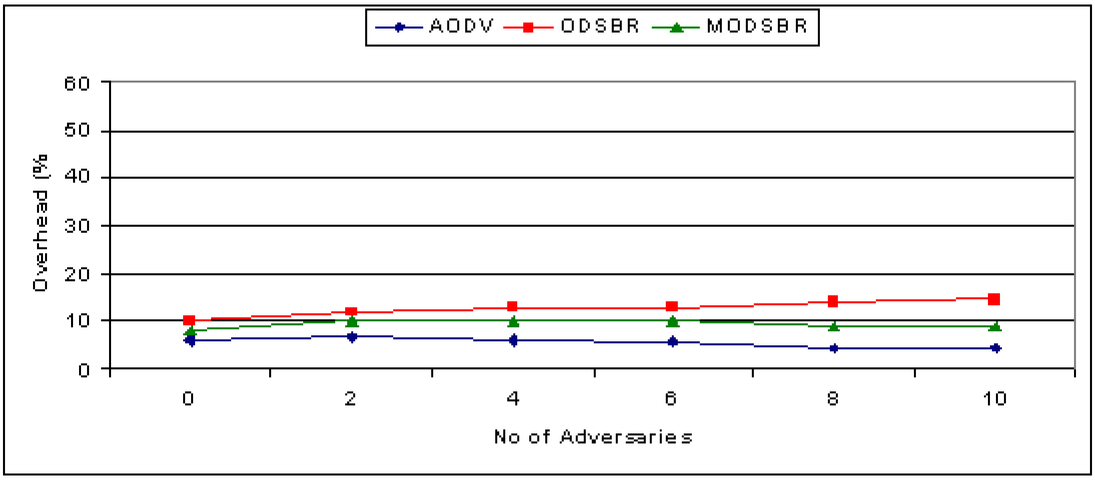


Figure 13. The overhead for the ODSBR and MODSBR in the presence of the black hole attack attacks

Figure 14 shows the overhead of ODSBR, MODSBR, and AODV with the presence of super wormhole attack. As shown the overhead of the MODSBR has decreased less than the ODSBR, but still greater than AODV. This is because disabling route caching is not the only reason for the high overhead.

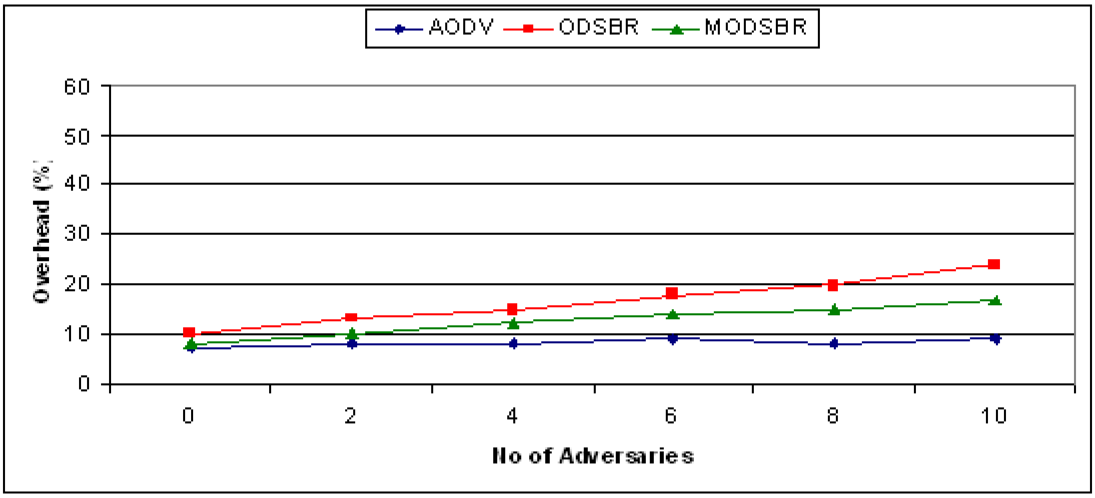


Figure 14. The overhead for the ODSBR and MODSBR in the presence of the super wormhole attack attacks

To conclude the previous results, we have listed the average of the overhead values in Table 12. Simulation experiments showed that the MODSBR outperforms ODSBR by an average of 23% decrease in routing overhead. To be noted that %average overhead in table 12 is calculated according to %average = (AVGODSBR - AVGMODSBR) / AVGODSBR.

	ODSBR	**MODSBR**	**% Average overhead**
Normal	24.5	20	% 18.3
Black hole	12.8	9.3	% 27.27
Wormhole	16.7	12.7	% 24.
Averages	18	14	**% 23.216**

Table 12. Average Overhead Values

6. Conclusions

ODSBR routing protocol is a very effective secure on-demand routing protocol that is resilient to Byzantine failures. But disabling the route caching property is a factor that causes decrease in the performance of the ODSBR protocol. So In order to enhance the performance of the ODSBR routing protocol, ways of taking advantage of route caching were investigated. The performance of the MODSBR protocol was analyzed in the presence of adversarial scenarios and in the normal behavior (non-adversarial scenarios). Our experiments showed that MODSBR outperforms ODSBR by an average 23% decrease in the overhead.

Another technique was implemented to MODSBR to improve its ability for security called "Central Clustered Link Weight Management". This technique aims to make nodes know about links that caused network failure faster than previously. This was achieved by specifying a set of Supervisory nodes to store a general weight table, where any node can use this table during computing the total weight to any path or to update the weight for any link. The impact of the Byzantine attacks on the MODSBR was evaluated by computing the percentage delivery ratio (%DRE). The results were compared with the original ODSBR. The results showed that the MODSBR protocol was able to increase the delivery ratio in the presence of the black hole attack with and without flood rushing. Also it has increased the delivery ratio in the presence of the wormhole attack for the random placement case and the super wormhole attack for the complete coverage case. But in the presence of the central wormhole attack or the cross of death attack the delivery ratio was increased only for high mobility while for low or no mobility the original ODSBR is better to be used. On the average, we can say that MODSBR had improved the delivery by approximately 10% on the average.

7. References

[1] Awerbuch, B. Holmer, D. Nita-Rotaru, C., and Rubens, H. “An on-demand secure routing protocol resilient to Byzantine failures,” in ACM Workshop on Wireless Security (WiSe), September 2002.

[2] Awerbuch, B. Holmer, D. Curtmola, R. Nita-Rotaru, C., and Rubens, H. “Mitigating Byzantine Attacks in Ad Hoc Wireless Networks,” Technical Report Version 1, March 2004.

[3] Awerbuch, B. Holmer, D. Curtmola, R. Nita-Rotaru, C., and Rubens, H. “On the Survivability of Routing Protocols in Ad HocWireless Networks,” in the First International Conference on Security and Privacy for Emerging Areas in Communication Networks (SecureComm), 2005.

[4] B. R. Smith, S. Murthy, and J. Garcia-Luna-Aceves, “Securing distance-vector routing protocols,” in Symposium on Networks and Distributed Systems Security, 1997.

[5] C. E. Perkins and E. M. Royer, Ad hoc Networking, ch. Ad hoc On-Demand Distance Vector Routing. Addison-Wesley, 2000.

[6] C. Perkins, E. Belding-Royer, and S. Das, Ad hoc On-Demand Distance Vector (AODV) Routing. IETF – Network Working Group, The Internet Society, July 2003. RFC3561.

[7] Hauser, R. Przygienda, T., and Tsudik, G. “Reducing the cost of security in link-state routing,” in Symposium of Network and Distributed Systems Security, 1997.

[8] Hu, L., and Evans, D. “Using directional antennas to prevent wormhole attacks,” in NDSS 2004, 2004.

[9] Hu, Y, -C. Johnson, D, B., and Perrig, A. “SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks,” in The 4th IEEE Workshop on Mobile Computing Systems and Applications, June 2002.

[10] Hu, Y, -C. Perrig, A., and Johnson, D, B. “Ariadne: A secure on-demand routing protocol for ad hoc networks,” in The 8th ACM International Conference on Mobile Computing and Networking, September 2002.

[11] Hu, Y, -C. Perrig, A., and Johnson, D, B. “Packet leashes: A defense against wormhole attacks in wireless ad hoc networks,” in Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003), April 2003.

[12] Hu, Y, -C. Perrig, A., and Johnson, D, B. “Rushing attacks and defense in wireless ad hoc network routing protocols,” in ACM Workshop on Wireless Security (WiSe), 2003.

[13] Kent, S. Lynn, C., and Seo, K. “Secure border gateway protocol (s-bgp),” IEEE Journal on Selected Areas in Communication, vol. 18, no. 4, 2000.

[14] Marti, S. Giuli, T. Lai, K., and Baker, M. “Mitigating routing misbehavior in mobile ad hoc networks,” in The 6th ACM International Conference on Mobile Computing and Networking, August 2000.

[15] Papadimitratos, P., and Haas, Z. “Secure data transmission in mobile ad hoc networks,” in 2nd ACM Workshop on Wireless Security (WiSe), 2003.

[16] Papadimitratos, P., and Haas, Z. “Secure routing for mobile ad hoc networks,” in SCS Communication Networks and Distributed Systems Modeling and Simulation Conference, pp. 27–31, January 2002.

[17] Sanzgiri, K. Dahill, B. Levine, B, N. Shields, C., and Belding-Royer, E. “A secure routing protocol for ad hoc networks,” in 10th IEEE International Conference on Network Protocols (ICNP’02), November 2002.

[18] “The network simulator - ns2.” http://www.isi.edu/nsnam/ns/