Contradiction Based Gray-Hole Attack Minimization for Ad-Hoc Networks

Nadav Schweitzer, Ariel Stulman, Member, IEEE, Asaf Shabtai and Roy David Margalit

Abstract—Although quite popular for the protection of ad-hoc networks (MANETs, IoT, VANETs, etc.), detection & mitigation techniques only function after the attack has commenced. Prevention, however, attempts at thwarting an attack before it is executed. Both techniques can be realized either by the collective collaboration of network nodes (i.e., adding security messages to protocols) or by internal deducation of attack state. In this paper we propose a method for minimizing the gray-hole DoS attack. Our solution assumes no explicit node collaboration, with each node using only internal knowledge gained by routing information. The technique was evaluated using 5 different threat models (different attacker capabilities), allowing for a better understanding of the attack surface and its prevention. Our simulation results show a decrease of up to 51% in previously dropped packet, greatly minimizing gray-hole attack effectiveness.

Index Terms—MANET, OLSR, Gray-Hole Attack, DCFM.

I. INTRODUCTION

With the growth in the use of MANETs, as a stand-alone networking tool and as the basis for other emerging technologies such as IoT and VANETs [1] the demand for security on this underlying technology is increasing as well. Ubiquitous MANET protocols (i.e., AODV [2], DSDV [3], OLSR [4], etc.), however, were developed with the focus on efficient routing and data transfer performance, not security issues. This, in turn, led to the current situation where these protocols are vulnerable to a multitude of attacks, including spoofing attacks [5], flooding attacks [6], wormhole attacks [7], replay attacks [8], [6], black-hole attacks [9], colluding mis-relay attacks [10], and many others.

Black-hole attack [9], and the more general gray-hole attack [11], on MANETs, are manifested when a malicious node is able to silently discard some (gray-hole) or all (black-hole) of the messages passing through it. The attack can be further amplified if the attacker is able to cunningly manipulate routing tables so as to increase the probability that messages would be routed through it. Of the two attacks, gray-hole is more devastating and of higher sophistication, as it selectively discards messages, making detection and/or avoidance difficult [12], [13]; rendering anti-black-hole algorithms such as [14] useless [15]. Thus, mitigation of the gray-hole attack will also solve the more famous black-hole attack as well.

Denial Contradictions with Fictitious Node Mechanism (DCFM) [16], is an algorithm devised to specifically address a denial of service (DoS) attack variant called node isolation [4] in OLSR based networks. DCFM’s main virtues are its ability to mitigate the node isolation attack by relying solely on internal knowledge acquired by each node during routine routing and in utilizing the same technique used for the attack to prevent damage. As both node isolation and gray-hole attacks require similar preliminary steps for attack execution, namely coaxing a victim into appointing the attacker as sole multi-point relay (MPR) node, which is responsible for broadcasting a node’s existence to the network (see section II-A for further details), we found DCFM to be a good basis for mitigating the gray-hole attacks as well. As it turns out, although being a sole MPR isn’t a requirement for gray-hole attacks to commence (albeit, without guarantee, as choosing a path through the attacker is equally viable as other alternative paths - see Appendix), the information provided by DCFM can be used to minimize it as well. These techniques, dubbed IMP (short for IMPovement), were implemented in the NS3 [17] simulator, with results showing an improved detection rate of up to 51% of all previously dropped packages.

The reminder of this paper is organized as follows. In Section II-A the OLSR protocol is presented; then, the gray-hole attack is described. In Section II-C we briefly present the DCFM algorithm. The method for protecting OLSR MANET from gray-hole attack using DCFM is described in Section III. Section IV describes the simulation model and presents the results achieved along with a discussion of the results. Previous works related to OLSR security as well as to the gray-hole attack are discussed in Section V. Finally, conclusions and future works are presented in Section VI.

II. BACKGROUND

A. OLSR overview

The Optimized Link State Routing protocol (OLSR) is a proactive routing protocol designed for large and dense networks, and highly recommended for VANETs as well [1], [18], [19], [20], [21]. The OLSR protocol is an optimization of the classical Link-State Routing protocol (LSR), aimed at reducing network overhead. While the original LSR uses a flooding propagation technique in which a node receiving any message must re-transmit it to all its neighbors, OLSR selectively re-transmits messages based on a specified set of rules. The crux of the optimization is based upon a subset of 1-hop neighbors, called multi-point relays (MPR), which are designated as forwarding agents for control packets throughout the network.

MPRs are selected by a node as a subset of its 1-hop neighbors, such that the MPR set allows coverage of all of its...
2-hop neighbors. By minimizing its MPR selections, a node is able to transmit messages to all 2-hop neighbors with minimal duplication. Thus, both topology control messages and data packets are only forwarded by this minimal MPR set, allowing for fewer duplicate messages while maintaining network-wide coverage.

There are two types of messages used to discover network topology in OLSR: HELLO and TC (i.e., topology control). The HELLO message, which declares a node’s knowledge of its surrounding, is broadcast to all neighboring nodes. Any node that can hear the broadcast and reciprocate back to the sender is classified as a 1-hop neighbor. Consequently, each node acquires its local topology up to a 2-hop range.

In addition, OLSR requires that all nodes selected as MPRs periodically advertise a TC message listing all nodes that have selected the sender as its MPR. These control messages are only propagated through the MPR super-network, reducing overall network traffic.

Each node in the network maintains network topology based on both the HELLO and TC messages it receives. It then calculates and stores, for each node discovered, the shortest distance (i.e., the minimal required hops between the source and the destination) between itself and one of the destination’s node MPRs; hence, the shortest path to the destination. See [22] for more details.

B. Black- and Gray-hole Attack

Black holes in the network refer to locations where malicious nodes discard network traffic without the source being notified that the packet did not reach the requested destination. Regardless of the mobile routing protocol, every node on the path between the source and destination is a potential black-hole attacker. The attack surface can be enhanced, however, with specific steps executed by the attacker to increase the probability of landing on the path to/from a specific (or all) victim(s). Thus, our main concern with black-holes, albeit not the only concern, is when a node can illegitimately coerce the topology so as to be placed on the path between the victim and some other node, more than the random chance of such an occurrence.

Black-hole is a special case of the more general gray-hole, in which packets are selectively dropped while allowing others through. In this paper we focus on the case in which the attacker selectively forwards data packets of every node except the victim’s. It does not try to isolate the victim; thus, control packets are forwarded.

An OLSR based network is vulnerable to gray-hole attack. The attacker may send, for instance, a bogus HELLO messages to its 1-hop neighbors, claiming to know more 1-hop neighbors than it actually does. This will illegitimately increase its probability of being chosen as a sole MPR by its neighbors. The more neighbors an attacker claims to have, the larger the potential impact of the attack.

Consider Figure 1, depicting a specific network topology, where \( x \) is an attacker and \( v \) a victim. \( x \) advertises a bogus HELLO message containing \( \{f, v, g\} \); namely, \( v \) and each of its 2-hop neighbors, and adds a fictitious \( F_x \) in order to ensure the attack’s success. Being the most cost-effective node in \( v \)’s view of the network topology, it nominates \( x \) as its sole MPR. From here the attack can easily commence, as nodes from all around the network will direct data traffic destined for \( v \) towards \( x \), who can drop packets at will.

C. Denial Contradictions with Fictitious Node Mechanism (DCFM)

DCFM was proposed by [16] in order to address the problem of node isolation in OLSR based networks. It identifies potential malicious nodes trying to falsify HELLO messages using only internal information within the victim, without relying on any centralized or external trusted party. Such early detection prevents a possible attack before it can manifest. DCFM verifies the validity of a HELLO message by looking for contradictions between what the message claims and its pre-acquired topological knowledge. According to DCFM, sole MPRs nominations are allowed only when no contradictions are found. With the presence of contradictions, an MPR can be nominated for all 2-hop neighbors for which the suspected node is the only access point. It cannot, however, be nominated as sole MPR for 2-hop neighbors that can be reached through other paths. Following [23], and as justified in [16], in this work we assume that TC messages cannot be spoofed.

1) notation: Following [16] we use the notation below for the remainder of this work:
   - \( V \) denote the set of all nodes in the network,
   - \( v, x \in V \) are the victim (as well as/or the receiver) and attacker nodes, respectively,
   - \( F_x \) is a fictitious node advertised by \( x \).
   - \( ADJ(v) \subseteq V \) is the set of all 1-hop neighbors of \( v \),
   - \( ADJ_2(v) \subseteq V \) is the set of all 2-hop neighbors of \( v \),
   - \( MPR(v) \subseteq ADJ_1(v) \) is the set of 1-hop nodes of \( v \) who appointed \( v \) as their MPR, and
   - \( MPRI(v) \subseteq ADJ(v) \) is the set of 1-hop nodes who were selected by \( v \) as MPRs.

2) contradiction rules: DCFM defines three rules that must be satisfied before a HELLO message sender is considered trustworthy. Only trusted senders can be nominated as sole MPRs for 2-hop nodes that can otherwise be reached, subject
to the OLSR protocol. A detailed explanation of these contradiction rules and their inherent logic can be found in [16]:

1) When node $x$ advertises a HELLO message containing $\text{ADJ}(x)$. For every node $z \in \text{ADJ}(x) \cap \text{ADJ}(v)$, $v$ should verify that $x \in \text{ADJ}(z)$.

![Figure 2. Sample node topology for identifying contradictions](image)

Rule No. 1 can be explained by Figure 2 in which $\text{ADJ}(v) = \{x, u, z\}$ and $x$ is an attacker. $x$ advertises a HELLO message claiming to know the set of $\text{ADJ}(v)$ containing $z$ (since $z$ is a $v$’s 2-hop neighbor through $u$). However, $z \notin \text{ADJ}(y)$ and since $z$ has not included $x$ in its HELLO message, $v$ suspects $x$.

2) For each node $y$ mentioned in a HELLO message, $v$ should check whether there exists $z \in \text{ADJ}(y)$, such that (a) $z \notin \text{ADJ}(x)$; hence, not mentioned in $x$’s HELLO message and (b) $y \in \text{ADJ}(v)$; thus, $z$ is located at least 3-hops away from $v$. Once these conditions are fulfilled.

3) $v$ must treat a HELLO message containing all nodes of the network except for $\text{ADJ}(v)$, as a potential attack. Nodes must apply each of the mentioned rules sequentially, advancing from one rule to the next iff there are no contradictions. Failure of any of the rules would require that $v$ appoint $x$ as a sole MPR only for the nodes that were exclusively declared in its HELLO message.

For more details about DCFM, the interested reader is referred to [16] where this claim is corroborated.

III. PREVENTING THE GRAY-HOLE ATTACK USING DCFM

The original DCFM was developed in order to identify and prevent the node isolation attack. In the gray-hole attacks, however, this solution is incomplete. Attackers can still orchestrate their attack by dropping data packets that were to be routed through them – even when they were not appointed as sole MPRs.

![Figure 4. Illustrating the same network of Figure 1 with the protection of DCFM](image)

Avoidance of selecting a suspected node as a sole MPR, which is the crux of DCFM, mainly prevents the gray-hole attack. There are, however, two additional venues in which a malicious node can circumvent DCFM based protection: (1) when it is a natural candidate for passing data from $\text{ADJ}(v)$ to $v$; and (2) when topology restraints require that it be appointed as sole MPR, i.e., when there is no other connection to some node. Our simulations show that although the probability of attack success is less in either of these attack venues when compared to the main venue, nonetheless, it is still feasible. Using internal knowledge gained by DCFM, we present an improved method denoted by IMP (short for IMPovement), as a method of further decreasing attack success to include these two venues as well.

Figure 4 illustrates the same network of Figure 1, with fictitious nodes advertised by $x, d, g, s, a, e, s, x, g$, and $h$ in accordance with the fictitious node mechanism of DCFM [16]. $x$ has not nominated any MPR for covering $\{s, F_g\} \in \text{ADJ}(g)$ despite $x$’s (false) claim that $g \in \text{ADJ}(x)$; hence, $\{s, F_g\} \in \text{ADJ}(x)$. Using DCFM rules, $v$ identifies the malicious intent of the attacker $x$, and refrains from nominating $x$ as a sole MPR; thus, $\text{MPR}^R(v) = \{a, i, x, e\}$.

We point out that although $\text{ADJ}(x) = \{v, f, g, F_a, F_e, F_s\}$, or so it (falsely) claims, $\text{MPR}^R(v) \neq \emptyset$, as others are also appointed into $\text{MPR}^R(v)$. More importantly, however, it must be noticed that $x \notin \text{MPR}^R(v)$, in spite of being suspected. This is required for accessing $\{F_s\}$ the set of nodes advertised exclusively by $x$, and is essential for MANETs initialization and further evolution. Now $x$ is in the position of attacking $v$ despite DCFM’s protection, using one of the supplemental attacks described above.
Consider the case, for the topology depicted in Figure 4, where $h$ sends a message to $v$. Since the shortest path goes through $f$, $h$ delivers the message to $f$. As $f \in ADJ_2(v)$, it gets to choose a path going through one of the three nodes $x$, $n$ or $e$ irrespective of who was appointed MPR for $v$. That is, the probability that packets travel to the victim through the attacker is still high.

To deal with these problems we propose using DCFM’s contradiction rules to further influence routing decisions. Not only will we decide who should be in $MPR^r(v)$, but other nodes in the network also make data routing decisions – on the fly – based on the previous outcomes of the rules. We call this improvement IMP, which can be summarized by Algorithm 1 in which $k$ is a node on the optimal path between the source and destination nodes, and $d \in ADJ_2(k)$ located further down the path.

**Algorithm 1 IMP**

```plaintext
IMP(k,d)
S ← (ADJ(k) ∩ ADJ(d))
for each x ∈ S do
  mark x as suspect or legitimate using Algorithm 1 of DCFM [16, Section 3]
  if x marked as legitimate or |S| = 1 do
    return x
else
  S ← S - {x}
```

In Figure 4, besides $v$, $f$ will also suspect $x$ as malicious. In $f$’s view of the network, $x$’s HELLO message contains a contradiction. $x$ is advertising that $ADJ(v) = \{v, f, g, F_a, F_v, F_e\}$. Rule No. 2, however, proves to $f$ that $x$ is lying and $g$ isn’t a neighbor of $x$. $f$ will refrain from routing data through $x$ if other suitable options exist (e.g. nodes $n$ or $e$). Of course, lacking better options, nodes would must still route data through suspicious nodes.

Simulation found that in some cases this route choosing mechanism has increased the number of delivered packets by 20% beyond what was achieved using simple DCFM; this, without any additional cost.

As augmenting DCFM with IMP for preventing the gray-hole attack does not increase network overhead beyond what has been discussed in [16] for DCFM, we claim that the networking cost if IMP is negligible.

**IV. EVALUATION**

Since network topologies are infinite, expected benefit estimation must be achieved through simulation. In this section we describe the varied simulations that were ran in order to justify the using of IMP for preventing gray-hole attacks.

**A. simulation**

We used the built-in OLSR module in the network simulator NS3 [17]. It was augmented to run DCFM in accordance with the protocol above. All simulation value sets were run ~1000 times, with values reported as averages over these results. The movement, where relevant, was 1.5-2 m/s (5.4-7.2 km/h). The transmission range was about 250 meters.

A set of simulations was designed to test the effectiveness of DCFM against gray-hole attacks. For this purpose we used a random network topology with a varying number of nodes, fluctuating network density from 30 to 100 nodes in an area of 750x1000m. A simulation round without connectivity between its components was discarded, and was not taken into account in the calculated outcome.

In each simulation, three predefined nodes were used: a victim, an attacker, and a source node used for sending messages to the victim. Both the victim and the source nodes were randomly placed. We require, however, that victim and source must be at least 2-hops from each other, discarding simulations not satisfying this requirement. Justification of this constraint is based on the observation that victim and source that are 1-hop neighbors are inherently protected from attackers, and renders all other protection superfluous.

The attacker was designed with one of the 5 following different capabilities:

I. **Passive Silent attacker (PSV).** This attacker was randomly placed within the network. It has done nothing for increasing its chances of becoming a routing node for the packets (in order to drop them). Results of this attacker type were used as a baseline for the gray-hole attack when compared with the more sophisticated attacks.

II. **Randomly located attacker (RND).** Similar to the passive attacker, this malicious node is randomly placed within the network. It differs by the fact it would try to get itself appointed as a sole MPR of the victim whenever they are 1-hop neighbors.

III. **Initially 1-hop neighbor attacker (1HOP).** Attacker who is initially located as a 1-hop neighbor of the victim. This attacker is similar to the one above, except its initial position isn’t random. It is purposely placed close enough to the victim so as they will begin as 1-hop neighbors.

IV. **Shadow attacker (SHDW).** This attacker was given the capability of shadowing the victim’s movements from a distance of 190 meters, constantly remaining a 1-hop neighbor of the victim. This distinguishes it from the previous attacker who only begins as a neighbor, but the distance can increase as the simulation commences.

V. **MITM attacker (MITM).** This attacker, improves the ability of the shadow attacker. Not only does it remain a 1-hop neighbor poised for attack, it is given awareness for the source node location. This allows it to locate itself on a line between the two nodes, increasing the likelihood of being on the shortest path between the source and victim.

For each of the attackers, we examined the follow cases:

(a) The package arrived at its destination (**arrived**). 
(b) The package was lost by third party on its way for some obscure reason irrespective of the attacker (**lost3rd**).
(c) The package was dropped by the attacker, who (by chance or orchestrated) is a neighbor to the victim, even though there was at least one other node who could have forwarded the packet (**attackerNeighbour**).
(d) The package was dropped by the attacker, who (by chance or orchestrated) is a neighbor to the victim, but was the only route available (attackerSingleNeighbor).
(e) The package dropped by the attacker located at least 2-hop from the victim (attacker). We hypothesized that attackerNeighbour can be mostly influenced by IMP lost3rd and attacker should not be influenced beyond some random, independent change. No improvement should be noticed for attackerSingleNeighbor, as IMP cannot change network topology; merely, choose wisely between equivalent paths. With a single path to the victim, there are no alternate options that circumvent the attacker for IMP to choose from.

Each simulation was run without any attack, under attack without any protection, under attack while under DCFM protection and, under attack while under IMP protection. This was repeated both with and without movement of the participating nodes.

B. results

Figures 5 and 6 show the percentage of dropped messages by each attacker for different population sizes in networks with movement (M) and without movement (NM), respectively.

As can be seen in both graphs the attackers’ maximum success is when the population density is lowest (30). In addition, as expected, The MITM attacker is strongest while the PSV attacker is usually the weakest.

For this reason we developed IMP – the improvement of DCFM. Simulations show that even in the relatively limited damage of the attacker, IMP increases the number of the received messages and minimizes the damage caused by the attack. Moreover, the sparser the network, the greater the impact of IMP.

Figures 7-16 present the percentage of the arrived messages with attack (UA) and without attack (NA), under attack with the protection of simple DCFM (DCFM) and under attack with the protection of IMP (IMP), with (M) and without (NM) movement of the participating nodes for each of the attacker groups. In each figure, The X-axis represents the number of nodes in a random network topology – from 30 to 100 – while the Y-axis represents the percentage of arrived messages in each of the scenarios.

C. improvement analysis

To better understand the improvement obtained by IMP, we analyzed the different causes for dropped packets (see IV-A options (b)-(e)). This allowed us to pinpoint the decrease of dropped messages due to IMP as a percentage of all previously
dropped packets. We justify this technique so we can check whether IMP improved the results, or some other independent external factor (i.e., lost3rd or attacker).

The bar-chart of Figure 17 depict a dissection of discarded messages in a 30-node network without movement. For each attacker class (the PSV attacker isn’t shown as its results cannot be influenced by IMP) we examine the results when under attack (UA) without any protection, with DCFM active (DCFM), and when IMP is active as well (IMP). The different reasons for un-successful communication ((c)-(e)) are shown. In addition, we also show the percentage of packets that do arrive with DCFM or IMP in place, which denotes the percentage of saved packets (SP); in essence, the successful improvement of this protection technique.

When comparing the results, one can deduce that although DCFM improves on the arrival ratio, IMP further improves this success percentage. And although DCFM improvements start at a low of 1% for the MITM attacker to a high of 27% for the SHDW attacker, IMP improvement are never lower than 30% (for the 1HOP attacker) and reach a maximum of 51% for the SHDW attacker.

It is noteworthy to mention that the majority of saved packets come from attackerNeighbour, as IMP tries and has the opportunity to circumvent the attacker due to the existence of an alternative path for forwarding packets.

Similar data is shown in Figure 18; a 30-node dense network that allows random movement (M) based on the rules described in IV-A. Again, we ignore the PSV attacker as IMP cannot (and does not) influence its results. We do, however, include lost3rd as one of the causes for dropped packets. Node movement create spontaneous communication outages, lost links, etc. which are valid reasons for losing messages.

For each attacker using IMP there is a noticeable (7% for the RND attacker to 30% for the SHDW attacker) decrease of dropped messages, where most of the improvement came from attackerNeighbour – as expected. This is in stark contrast to DCFM where besides a maximum of 17% improvement ratio, we can also notice a 6% increase of dropped messages for the RND attacker (denoted as a negative number of saved packets (SP)).

The rest of the figures show other sample results, for different node densities with (M) or without (NM) movements.

V. RELATED WORK

Black holes are well studied for AODV routing protocol [24], and are achieved by having a malicious node reply to every route request it hears. It then drops packets that are routed through it, creating a black hole in the network.

In [25], the authors suggest an intrusion detection system (IDS) in order to prevent selective black hole attacks at an AODV protocol based network. Their solution assumes several IDS nodes are deployed in MANETs in order to detect and prevent the attacks. Each of these nodes has to continually sniff all routing packets within its transmission range. This solution is problematic in terms of MANET because it assumes that there are third-parties in the network and the overhead is high.
In addition, in case one of the IDS nodes is malicious itself, nothing can prevent the attack.

TOGBAD, a solution proposed by Gerhards-Padilla, et al. [9], uses a central node to thwart attacks. Sensors placed throughout the network help in creating a central topology graph, with a neighbor count stored for every node. This count is then compared to the number of neighbors each node claims to have. A node will be declared malicious if the difference is greater than some predetermined threshold.

This solution requires sensors throughout the network, and some central, all-knowing master node. Both requirements counter MANET’s declared purpose; namely, that networks can be setup ad-hoc. These networks must be preconfigured with trustworthy sensors and must have all nodes succumb to a central trusted third party. IMP avoids these requirements. In addition, TOGBAD opens up a new attack venue, as nodes can cause others to be declared malicious by forwarding false information to TOGBAD regarding some potential victim.

[26] proposed to use security monitoring nodes (SMNs), a set of trusted nodes, as a solution against cooperative black hole attack in mobile ad-hoc networks. SMNs are activated only if the comparison between a message sequence number and a threshold value, computed by the expected packets and their corresponding ACKs passing between the source and destination nodes, is abnormal. Once a suspicious node is detected, the SMNs broadcast this knowledge to the network.

This solution suffers from the same problems of TOGBAD; mainly, the assumption that trusted nodes exist in the MANET and the introduction of an ALARM packet which itself is a security vulnerability.

[27] deals with a cooperative black hole attack. By adding two control packets called 3-hop ACK and HELLO-rep, a node can verify whether a candidate node is legitimate or not. The solution requires that each MPR node know its 3-hop neighbor set, in order to be able to discern whether a malicious node, sitting out of its transmission range, misuses transmitted TC messages. It also requires that each MPR node forward all TC messages from its MPR selectors even if multiple copies of the message are received. As a result, a node can distinguish whether a TC message is dropped intentionally.
by a malicious node or by a legitimate node just because of the duplication of two TC messages. This solution is flawed as two consecutive attackers (one in \textit{ADJ2} and the other in the 3-hop neighbor set) cannot be detected. And although the results show a high detection rate under various scenarios, the required communication overhead is substantial.

In [10], Kannhavong et al. attempt to mitigate the problem of colluding attackers. By modifying the HELLO message to include all 2-hop neighbors, a node can detect existing contradictions between messages, thus identifying an attack. Of course, as the authors themselves noted, it is difficult to distinguish between contradictions which occur due to an attack as opposed to those resulting from topology changes. In addition, such contradictions identify an attack but fail to identify the culprit.

Raffo et al. [5] propose a mechanism to improve the security of the OLSR routing protocol against external attackers. In their solution, each node signs its HELLO and TC messages. These signatures are later used by others to prove their own HELLO and TC messages. The resulting solution prevents devices from declaring imaginary links with known nodes. This solution functions correctly preventing spoofing links but fails where (1) the attacker is a natural candidate for passing data or (2) the attacker was nominated as an MPR for some other node. In those two cases the attacker still can drop packets going through it. IMP, however, checks whether a particular node is suspected, and eliminates the possibility of the package passing through it.

In addition, their mechanism is expensive in terms of additional overhead: signing the messages requires extensive computation, a cumulative factor that grows as the size of the network increases. Another problem is the fact that the network loses its spontaneity as all nodes are required to know each other in advance in order to share their public keys. This prevents the network from naturally evolving from the various nodes that appear at a certain place and time, a fundamental trait of MANETs.

Chen et al. [20] propose a method to create a trusted routing on VANET. Although their method can help in the prevention of illegal access, impersonation attack, link spoofing etc., it cannot help in the case where a legitimate node turns malicious after authentication. Their approach demands prior knowledge of network players, a demand which cannot be guaranteed in ad-hoc networks and should not be required for VANETs either. Finally, their solution requires high network overhead.

The authors of [28] claim that the damage caused by the attack of black- and gray-hole to MANET based OLSR protocol is less significant when compared to the damage caused by the same attackers in other protocols (e.g. AODV, DSDV). The authors check the Packet Delivery Ration (PDR), namely, the Ratio of sent data packets from source to data packets received at the destination. However, they do not take into consideration the lost packets (which actually have to be considered as infinity). As a result, long routes are not taken into account and therefore the results do not reflect reality. Another point is that they compare the results of OLSR, AODV and DSDV, and such comparison demands normalization since its PDR is better than the others.

Chang et al. [29] propose the cooperative bait detection scheme (CBDS) in order to detect and prevent a gray-hole attack in AODV based routing. By randomly selecting an adjacent node as its destination, a source node is able to bait malicious attackers into sending a reply RREP message; hence, flushing them out. This solution is interesting, but as the authors themselves write, it increases network overhead when under attack.

A heuristic approach, dubbed SNBDS, is proposed by Poongodi and Karthikeyan [30]. The approach suggests modification of the routing table structure for AODV based routing, among them: field status {normal/malicious/suspicious} for each node, field for the last RREP arrival time for the destination node that updated its sequence number. Together, these changes create a bait detection technique comparing the destination sequence number of the received RREP and that of the previously stored value. A database of suspicious nodes is composed, and propagated to all nodes using a RREQ message. Besides the shortcomings mentioned in their paper, it is assumed that all nodes are authenticated, a property which contradicts the basic ad-hoc premise of MANETs.

Another similar solution, proposed by Patel and Chawda [31], a threshold composed of three parameters: routing table sequence number, number of RREQs sent by the node, and the number of RREPs received by the node, is computed. During route discovery a RREP message with destination sequence number greater than the threshold value is ignored. Attackers, however, might behave correctly during the route discovery phase and only drop data packets transferred through it - a classical black-hole attack. To detect such behavior, [31] suggested that each node monitor packets transmitted by the next node down the route chain. A comparison of what it transmits vs. what the neighbor transmit would flush out a misbehaving black-hole attacker. An ALARM message with the malicious node identification is then broadcasted to all other participating nodes.

This solution is elegant, but contains some drawbacks. Clearly, it is only effective against a single attacker, but, as the authors note, fails when there are two consecutive colluding attackers. With the first attacker orchestrating the attack yet relying on the second to drop the packets, a listening benign node would have a proper count of outgoing packets. In addition, it deals with detection of attacks and not their prevention, allowing for an ALARM to sound only after the attack has manifested.

Extended data routing information (EDRI), proposed by [32] and improved by [33], is another technique used for attack detection. Every node maintains a personal EDRI table which contains the history of the packets send to and received from any neighboring node. A source node compares data entries of neighboring nodes with those of their neighbors, flagging mismatches as malicious; information which is then broadcast to the other participants in the network.

VI. CONCLUSIONS AND FUTURE WORKS

This paper presents an improvement algorithm for OLSR based networks (MANETs, IoT, VANETs, etc.) for mitigating gray-hole (and hence, black-hole) attacks. Using solely
internal knowledge gained by participating nodes, we are able to decrease captured packets by a double digit factor; well beyond what DCFM alone is able to accomplish under similar circumstances. Our only assumption is an active attacker trying to maliciously influence network topology to increase the attack surface. Although dormant attackers who can still go undetected can also drop packets, they cannot guarantee that routes will pass through them significantly decreasing the possibility of attack success.

For future planned work, following [1], [18], [19], [20], [21], we hypothesize that pending minor adjustments IMP can work for VANET environment as well.

VII. REFERENCES


Nadav Schweitzer received his bachelor’s degree in Software Engineering from the Jerusalem College of Technology, Jerusalem, Israel, in 2008. Later on (2012) he got his masters in Computer Science from Bar-Ilan University, Ramat-Gan, Israel. He is currently working toward the Ph.D. in the area of Security in Mobile Ad-Hoc Networks at the department of Information Systems Engineering at the Ben-Gurion University, Israel.

Ariel Stulman received his bachelor’s degree in Technology and Applied Sciences from the Jerusalem College of Technology, Jerusalem, Israel, in 1998. He then went on to get his masters from Bar-Ilan University, Ramat-Gan, Israel, in 2002. In 2005 he achieved a Ph.D. from the University of Reims Champagne-Ardenne, Reims, France. As of 2006 he holds a position at computer department of the Jerusalem College of Technology. His research interests are in the field of internet security and technologies, software testing, formal methods and real-time systems. Dr. Stulman is a member of the IEEE and ACM, and is the founding director of the cyber research group at JCT.

Asaf Shabtai is a lecturer at the Dept. of Information Systems Eng. at Ben-Gurion University. Asaf is a recognized expert in information systems security and has led several large-scale projects and researches in this field. His main areas of interests are computer and network security, machine learning, data mining and temporal reasoning. Asaf has published over 30 refereed papers. Asaf holds a B.Sc. in Mathematics and Computer Science (1998); B.Sc. in Information Systems Engineering (1998); a M.Sc. in Information Systems Engineering (2003) and Ph.D. in Information Systems Engineering (2011) all from Ben-Gurion University.

Roy David Margalit received his bachelor’s degree in Computer Science with distinction from the Jerusalem College of Technology, Jerusalem, Israel, in 2016. Currently a student towards M.Sc. degree in Computer Science at the Hebrew University, Jerusalem, Israel.