Liu, Gao; Yan, Zheng; Pedrycz, Witold

Data collection for attack detection and security measurement in Mobile Ad Hoc Networks

Published in:
Journal of Network and Computer Applications

DOI:
10.1016/j.jnca.2018.01.004

Published: 01/03/2018

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Data Collection for Attack Detection and Security Measurement in Mobile Ad Hoc Networks: A Survey

Gao Liu[a], Zheng Yan[a,b,*], Witold Pedrycz[c,d,e]

[a]*State Key Laboratory on Integrated Services Networks and School of Cyber Engineering, Xidian University, Xi'an 710126, China*
[b]*Department of Communications and Networking, Aalto University, Espoo, Finland*
[c]*Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6R 2V4, Canada*
[d]*School of Electro-Mechanical Engineering, Xidian University, Xi'an 710071, China*
[e]*Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia*

Abstract

Mobile Ad Hoc Network (MANET) is becoming one type of major next generation wireless networks. Nevertheless, it easily suffers from various attacks due to its specific characteristics. In order to evaluate and measure the security of MANET in real time and make this network react accordingly, a promising alternative is to integrate detection mechanisms that play a role of the second line of defense to detect attacks in MANETs. We note that in most attack detection mechanisms, it is essential and crucial to collect the data related to security for further analysis. If security-related data collection is untrustworthy, attack detection and security measurement might be impacted and disabled. Unfortunately, few existing studies concern security-related data collection in attack detection for the purpose of trustworthy security measurement. The literature lacks a thorough survey on security-related data collection for attack detection and security measurement in MANETs. In this paper, we propose a number of requirements for trustworthy security-related data collection, and then review detection mechanisms in MANETs that were published in recent 20 years. In particular, we employ the proposed requirements as a set of criteria to evaluate the existing work about security-related data collection. Based on the survey and evaluation, we identify a number of open issues and point out future research directions.

Keywords: MANETs, Security Measurement, Intrusion Detection, Data Collection

1. Introduction

MANETs allow wireless devices (called nodes) to communicate with each other in mobility through local wireless connections. There are two major communication scenarios in MANETs: 1) If two nodes are located at the transmission range of each other, they can employ their transceivers to exchange messages directly; 2) When two nodes cannot communicate directly, other nodes cooperatively help forward packets and these nodes are referred to as mobile routers. MANETs exhibit several specific characteristics. First, they have no fixed infrastructure. Second, nodes share common communication media (i.e., limited bandwidth). Third, the network topology of MANETs is dynamic due to a number of reasons (e.g., node mobility). Fourth, the battery power of nodes is constrained. Fifth, nodes have a low physical security level. Sixth, there is no management center. As a consequence, MANETs easily suffer from different kinds of security attacks.

*Corresponding author
Email address: zyan@xidian.edu.cn (Zheng Yan)

Security-related data collection for attack detection in MANETs becomes essentially crucial in order to evaluate and measure the real-time security of MANET and make this network react accordingly. It is promising to integrate various detection mechanisms that can be regarded as the second line of defense to detect attacks in MANETs. We note that in most attack detection mechanisms, it is essential to collect security-related data for further analysis. If security-related data collection is untrustworthy, attack detection and security measurement might be disabled. Mainly, untrustworthy security-related data collection will lead to distrustful detection results and wrong security measurement. For example, if security-related data are modified intentionally or unintentionally during their collection (i.e., generation, transmission, and storage), a false result could be produced. Unfortunately, many studies [1, 2, 3] of attack detection aim to design detection and reaction methods based on collected security-related data, but they seldom concern how to collect security-related data of high quality securely, efficiently and in a stable manner and achieve privacy at the same time. In addition, although there exist several surveys [4, 5, 6, 7] on attack detection, they mostly focus on the architecture, analysis and performance of detection methods and still neglect trustworthy security-related data collection.

In this paper, we study trustworthy security-related data collection in MANETs for the purpose of attack detection and security measurement. We first formulate a number of requirements in terms of trustworthy security-related data collection. Then, we offer a thorough review of detection mechanisms in MANETs, which were reported in recent 20 years. We employ the proposed requirements as criteria to evaluate the existing work about security-related data collection. Based on the survey and evaluation, we point out a number of open issues and propose future research directions. Specifically, the contributions of this survey are listed as follows:

- We propose the essential requirements of security-related data collection for achieving trustworthy attack detection and security measurement.
- We provide a holistic summary of the main attacks in MANETs and perform a thorough review of the detection mechanisms involving security-related data collection.
- We comprehensively summarize the types of security-related data that should be collected for the detection of various attacks in MANETs. Such data could serve as a significant reference for measuring the security of MANETs.
- We identify a number of open issues by employing the proposed requirements as criteria to evaluate the current literature.
- In particular, we point out that trust evaluation is missing in the existing literature with respect to security-related data collection for trustworthy attack detection and security measurement. This motivates our future research efforts.

The paper is organized as follows. Section 2 presents the requirements with regard to trustworthy security-related data collection. In Section 3, we first summarize the main attacks in MANETs. Then, we review single point detection mechanisms and intrusion detection systems against the main attacks. We evaluate the existing work about security-related data collection by employing the proposed requirements as evaluation criteria. In Section 4,

a number of open issues and future research directions are formulated. Finally, we conclude the paper in the last section.

2. Requirements on Security-Related Data Collection

In this section, we analyze some threats that might exist in security-related data collection, and correspondingly propose requirements behind trustworthy security-related data collection. In any detection mechanism, if the collected security-related data for detecting an attack fail to represent a real-time local scenario (e.g., fabricated data) or are not enough to ensure the accuracy of detection, the detection analysis yields an incorrect result, which disables the real-time security measurement of MANETs. Therefore, it should be ensured that the security-related data generated by a data provider fully reflect the real-time local scenario, this data provider transmits the data to a detector (i.e., data collector) securely, efficiently and in a stable manner, and this detector stores the data securely. In addition to those, some other threats (e.g., privacy disclosure) also should be taken into consideration since an attacker might set up future attacks by utilizing related vulnerabilities. As a result, some requirements can be elicited as follows.

2.1. Trustworthiness of Security-Related Data (TSRD)

The accuracy of attack detection and security measurement highly depends on the trustworthiness of security-related data [8, 9, 10, 11]. A data provider might offer false data due to objective circumstances (e.g., hardware failure). Possibly, the false data might be provided intentionally. For example, a compromised data provider mostly provides false data in order to avoid being detected. In many detection mechanisms, a positive detection result provided by a detector (e.g., a source or destination node) can trigger the reaction mechanism of networks. In some detection mechanisms, an alarm coming from a detector contributes to a final detection result, which, for example, can be determined via voting.

For simplicity, here we only consider the trustworthiness of the security-related data at the end of a data provider. It is worth stressing that these data can be categorized into two classes: the original data for the detection of a detector whose positive detection result triggers the reaction mechanism directly, and the alarm data that reflect the positive detection result of a detector and contribute to a final detection result determined by several detectors. Accordingly, we divide the trustworthiness of security-related data into the trustworthiness of original data (TO) and the trustworthiness of alarm data (TA).

2.2. Confidentiality (Co)

The confidentiality [12] of sensitive security-related data should be ensured, especially in a hostile environment. Some sensitive security-related data should not be revealed to any unauthorized parties, since an attacker attempts to employ the obtained sensitive data to launch future attacks.

2.3. Privacy (Pr)

Privacy [13] with regard to identity privacy, location privacy, and route privacy should be considered to avoid potential attacks.

As for the identity privacy (IPr), the identities of a source node and a destination node en route should not be revealed to anyone except themselves. In other words, both communication parties (i.e., the source node and the destination node) have an idea about the identity of each other, but any other nodes do not know the identities of these two parties. Additionally, the source node and the destination node should have no knowledge about the identities of any intermediate nodes en route.

With respect to the location privacy (LPr), the exact locations of source and destination nodes first should not be disclosed to anyone. Second, an arbitrary node, even an intermediate one, should not know the number of hops from the source node or the destination node. Generally, the identity and location privacy of the source and destination nodes should be ensured. For example, if an attacker knows there is a 1-hop distance between the source node and itself, directional antennas can be employed to determine the exact location of the source node such that the location privacy is violated seriously, and in a battle field even a physical attack might be launched.

In regard to the route privacy (RPr), three properties are required as follows. Firstly, anyone is not allowed to trace traffic flows to discover a source node or a destination node. Secondly, it is infeasible for anyone not existing en route to derive any part of the route. Finally, an attacker is not allowed to deduce the transmission and motion model of both the source and destination nodes.

2.4. Integrity (In)

As a basic security property, the integrity [14] ensures that security-related data cannot be modified when being transmitted and stored once generated. If the data are modified intentionally by an attacker when being transferred or stored, a detector inevitably might derive a false detection result, which ultimately impacts the security measurement of MANETs.

2.5. Authentication (Au)

A receiver is required to verify the source (i.e., data provider) of received security-related data [14]. If the receiver cannot determine the source, an attacker might employ this vulnerability and primarily might set up an impersonation attack, where the attacker impersonates a legitimate or authorized node to provide false data. Consequently, a detector might use the false data to execute detection analysis, which mostly could decrease the accuracy of security measurement.

2.6. Non-Repudiation (NR)

Anyone can verify truth if a repudiation occurs [12, 15]. Concretely, a receiver can ensure that anyone verifies if a node is the source of received security-related data. For example, after abnormal data have been recognized, a related data provider receives penalty (e.g., isolation), and thus the identified data provider should not be allowed to deny that it is not the data source. Based on the usage of security-related data, the non-repudiation is also comprised of two classes, i.e., the non-repudiation of original data (NRO) and the non-repudiation of alarm data (NRA).

2.7. Real Time (RT)

Real time reflects the high efficiency of security-related data collection [16]. In other words, a data collector receives security-related data as quickly as possible once a data provider sends them. The real time of security-related data collection assists the real-time security measurement of MANETs, and thus can help to reduce the loss introduced by security threats, attacks and intrusions as much as possible. After all, the requirement of real time allows MANETs to improve the performance in terms of self-purifying and prevention.

2.8. Stability (St)

The stability of security-related data collection refers to the ability that security-related data can reach a data collector from a data provider [17, 18]. There are some reasons why the data collector fails to collect the data, such as network congestion and the selfishness of data forwarder. Provided that sufficient data cannot arrive at the data collector, a computed detection result is not trustworthy, which negatively impacts the security measurement of MANETs.

2.9. Synchronization (Sy)

Synchronization [19, 20] requires that all involved data providers generate security-related data simultaneously for attack detection. If this property cannot be achieved, a detection result inevitably deviates from the true one.

3. Detection Mechanisms in MANETs

The attacks in MANETs can be categorized into two main classes: passive attacks and active attacks. In the passive attacks, attackers are honest but curious. Namely, an attacker does not disrupt any operations in MANETs, but attempts to derive valuable information by analyzing obtained data such as traffic flows. Nevertheless, the passive attacks (e.g., traffic analysis) can be eliminated by prevention mechanisms such as cryptographic techniques although it is not easy to discover them. With regard to the active attacks, they are the main attacks in MANETs and the commonly encountered approaches to defeat them are detection and reaction mechanisms. According to the Open System Interconnection (OSI) and Transmission Control Protocol/Internet Protocol (TCP/IP), we summarize the main attacks in MANETs as shown in Table 1, which are mainly reported in several mainstream academic research databases: ACM Library, IEEE Xplore Digital Library, Springer Library, ScienceDirect and Wiley Online Library. Then we review the related detection mechanisms published in recent 20 years in the above databases. In general, they can be divided into two categories, namely single point detection mechanisms and Intrusion Detection Systems (IDSs).

3.1. Single Point Detection Mechanisms

A single point detection mechanism is defined to detect one of main attacks. Here, we review existing single point detection mechanisms against main attacks in MANETs, which highly involve security-related data collection. We compare representative single point detection mechanisms in terms of when to perform security-related data collection, who to provide, forward and collect security-related data, how to collect security-related data, the type of security-related data in Table 2, and evaluate involved security-related data collection methods in Table 3 by employing the proposed requirements.

Table 1: Main Attacks in MANETs

Layers	Main Attacks	Description
Physical and MAC layer	Jamming attack	An attacker occupies the channel of victims to hinder their data transmission.
Network layer	Wormhole attack	To increase the chance of participating in a route, two attackers can establish a wormhole via in-band and out-of-band channel. In-band channel allows both remote attackers to collusively forward routing packets through honest nodes between them and presents a 1-hop distance. By using special hardware (e.g., cables), out-of-band channel can be established between two remote attackers and achieves low latency.
	Rushing attack	An attacker employs various methods to take part in a route with the shortest delay. These methods mainly include forwarding routing packets regardless of the delay of either MAC protocols or routing protocols, flooding a large number of bogus Route Requests (RREQs) to make the transmission queue of honest forwarders full, and high-power transmission.
	Black hole attack	Once becoming a part of a selected route, an attacker drops all received packets.
	Grey hole attack	As a part of a discovered route, an attacker drops packets selectively.
	Packet dropping attack	An attacker drops packets without an attempt to capture any route. This attack can be caused by low battery, overload condition, selfishness, or deliberately dropping packets.
	Sleep deprivation attack	An attacker might interact with a victim legitimately but aims to make the victim stay out of a sleep state, such that the victim's lifetime is reduced extremely.
	Sybil attack	Due to the lack of centralized authorities, an attacker might send packets by using either random identifications or the identifications of other legitimate nodes.
Transport layer	SYN flooding attack	An attacker sends a lot of synchronous (SYN) packets to a victim but responds no final acknowledgement, such that the backlog queue of the victim is filled with a lot of half open connection states and the victim thereby denies future connections.
	Man-in-the-middle attack	The man-in-the-middle attack allows two legitimate nodes to communicate via an attacker, and both of the legitimate nodes have no idea about the existence of the attacker. Therefore, the attacker can control the link between these two legitimate nodes.
Application layer	Worm attack	The worm is one of the most dangerous programs, which can duplicate itself and adopt the vulnerability of hosts. In the propagation of the worm, each copy employs an IP address to determine a next target, and a received worm code can be executed by this target. Hence, the worm attack brings some serious results such as the denial of system service.

3.1.1. Detection Mechanisms against Jamming Attack

In [21], Strasser et al. identified the reason of individual bit errors in a received packet to determine whether the packet is jammed or transferred in a poor channel. Concretely, there are external interferences if the bit error in a packet appears in an error sample acquisition and a relative Received Signal Strength (RSS) value is high, and the RSS value should be small if weak links introduce the error, which thereby can be employed to differentiate the packet errors of interferences and weak links. When a receiver directly collects packets and RSS from a sender (a neighbor of the receiver), a shared symmetric key between them is used to achieve the confidentiality. The integrity, the authentication and the non-repudiation of original data are achieved by exchanging the sequence generated by synchronized stream cipher. Moreover, the packets and RSS collection achieves the synchronization since the sender transmits each bit of packet via signal. However, the authors failed to consider other requirements (i.e., the trustworthiness of original data, the privacy, the real time and the stability). It is worth noting that the trustworthiness of alarm data and the non-repudiation of alarm data are not suitable for evaluating this security-

related data collection, because the packets and RSS can be considered as original data and the alarm data do not exist.

Xu et al. proposed a detection mechanism [22] by employing the principle that a jamming attack impacts the distribution of carrier sensing time (i.e., the amount of time a detector spends on waiting for an idle channel). First, when there is no jamming attack, a detector can obtain the distribution of carrier sensing time for a channel by using historical records or theories. Second, the detector senses the channel in real time and measures the carrier sensing time. Last, the detector compares the measured time with the distribution for identifying whether it is jammed.

After a receiver obtains packets from a sender, it can compute a Packet Delivery Ratio (PDR) expressed as P/M, where P is the number of the received packets passing cyclic redundancy codes check and M is the number of total received packets. The PDR can also help to detect the jamming attack. An experiment reported in [23] showed that a PDR measured can reach 78% even when congestion is introduced. The PDR, however, drops sharply when the jamming attack occurs. A poor link, a drain battery and other causes might introduce a false positive. To overcome these flaws, the receiver is allowed to measure RSS and PDR simultaneously. If the RSS is high but the PDR is low, there is a high possibility of the jamming attack.

In [24], Spuhler et al. proposed to compare an estimated PDR with a measured PDR from a sender for recognizing the jamming attack. More specifically, during the signal synchronization of packet transmissions, a receiver can estimate a PDR from a sender. Then, the receiver measures a PDR from the sender. Finally, it compares the measured PDR with the estimation result to identify the jamming attack. When a receiver obtains packets and computes an observed PDR, all the proposed requirements were not considered. Note that the PDR belongs to original data and thus it is not suitable to employ the trustworthiness of alarm data and the non-repudiation of alarm data to evaluate the PDR collection due to non-existent alarm data. Also, the synchronization should not be discussed since the receiver collects the PDR from a single sender instead of more senders for each time of detection.

The authors of [25, 26] considered the increase of the Euclidean distance (proportional to hop counts) between a source node and a destination node to detect the jamming attack. Before the detection of jamming attack, a source node has discovered multiple paths to the same destination node and obtained the shortest path for data packet transmission by computing and comparing the distance of these discovered paths. If the jamming attack occurs, an alternative route is chosen out of these paths to transmit packets and thus the hop counts of the used route increase, which also increases the distance and implies the presence of the jamming attack.

To detect the jamming attack, Hamieh and Ben-Othman [27] employed the fact that the access to a channel of a jammer depends on the access of active nodes and the dependence in a jamming state exceeds that encountered in normal networks. Concretely, a transmission node first collects reception error time and reception correct time via Media Access Control (MAC) protocol. Then, it computes the correlation coefficient value of the above collected data to quantify dependence. Last, the computed value is compared to a preset threshold value to identify the jamming attack.

To capture a channel, a DoS attacker modifies the IEEE 802.11 Distributed Coordination Function (DCF) standard and the MAC firmware code in the network interface card on its communication device. In order to detect this kind of attack, the authors of [28, 29] employed the fact that the number of packets transmitted successfully by

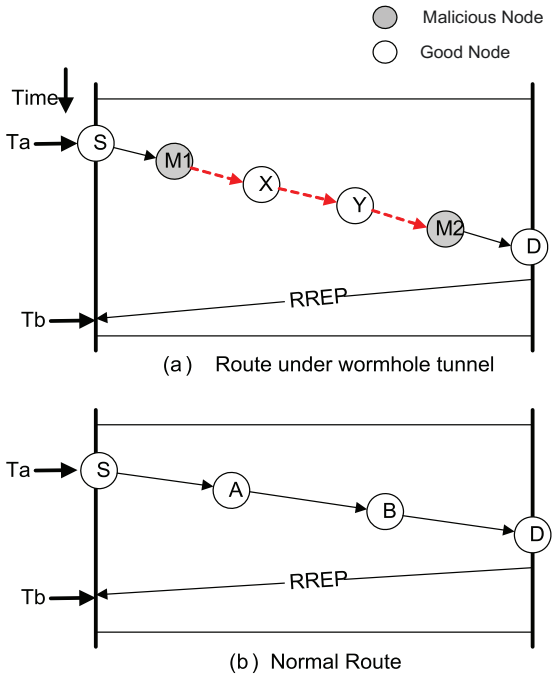


Figure 1: Route discovery delay [33]

a node is almost equal to the number of Clear-To-Send (CTS) packets received. Intuitively, during the data packet transmission phase, a node collects and counts CTSs from destination nodes. Then, the node determines a threshold value by studying the Markov chain. In addition, the node sorts the received CTSs according to destination MAC addresses, and finally a moving average can be computed and compared with the threshold value, which thus helps to identify a DoS attacker. However, during the collection of the number of CTSs, all the proposed requirements were not considered. The trustworthiness of alarm data and the non-repudiation of alarm data are not suitable for the evaluation of the collection since the number of CTSs does not belong to alarm data but original data and there exist no alarm data. Besides, the synchronization should not be taken into consideration, because a node constantly collects and counts received CTSs from a single destination node instead of more destination nodes for each time of detection.

A selfish node might misuse the IEEE 802.11 DCF to obtain more bandwidth, and thus might lead to the DoS attack. In distinctly greedy behaviors, it is most profitable to manipulate a backoff value (i.e., operate on a smaller Contention Window (CW) value) for increasing the probability of accessing a channel. Kyasanur and Vaidya proposed a detection mechanism [30] by using a shared backoff value between a sender and a receiver. First, a receiver informs a sender of a backoff value based on CTS and Acknowledgement (ACK). Second, in a next transmission, the sender adopts the received backoff value. Finally, the receiver counts the number of the idle slots between consecutive transmissions, and the sender is recognized if the number is less than a threshold value that is extracted from the backoff value. In the detection mechanisms presented in [31, 32], a sender and a receiver agree on a random number to ensure the fairness of accessing a channel.

3.1.2. Detection Mechanisms against Wormhole Attack

To detect the in-band wormhole attack, Choi et al. adopted the fact that for a RREQ or Route Reply (RREP) in Dynamic Source Routing (DSR) protocol, traveling a wormhole link is slower than traveling a normal link [33]. Given a distance TR that a packet can travel without forwarding, the propagation speed V_p of a packet, and the average velocity V_n of nodes, a Wormhole Prevention Timer (WPT) is denoted as $WPT = 2 * V_n * TR/V_p^2$. As shown in Figure 1, source node S sends a RREQ to destination node D at time T_a and receives a RREP from D at time T_b during the route discovery phase of DSR protocol. Therefore, after collecting the sending time and the receiving time, the source node computes a time delay per hop, i.e., $Delayperhop = (T_a - T_b)/Hopcount$, and the presence of a wormhole is confirmed if $Delayperhop > WPT$.

Statistical analysis of the end-to-end delay between a source node and a destination node can be employed to detect the in-band wormhole attack without the requirement of synchronizing their time. A source node sends a destination node a packet with the sending time. After receiving the packet, the destination node extracts the sending time and thus computes an end-to-end delay denoted as the difference between the receiving time and the sending time. Based on the parametric cumulative sum (P-CUSUM) [34, 35], the non-parametric cumulative sum (NP-CUSUM) [36] and the central limited theory [37], the collected sequence of end-to-end delays can be analyzed to recognize the in-band wormhole attack.

During the route discovery phase of DSR protocol, a secure routing protocol [38] requires a source node and a destination node to authenticate the delay of a RREP and a RREQ, respectively to recognize the in-band wormhole attack. First, a source node sends a destination node a RREQ with the sending time. Second, the destination node receives the RREQ and thereby computes a delay due to the sending time and the receiving time. Third, the destination node filters a duplicate RREQ whose delay per hop exceeds a cutoff value, and an accepted RREQ is employed to update the cutoff value. Similarly, the source node authenticates the delay of RREPs from the destination node to detect the in-band wormhole attack.

Packet leashes [39, 40] consist of a temporal leash and a geographical leash, which represent the expiration time of a packet and a maximum allowed distance of a packet from a sender, respectively. Therefore, packet leashes can restrict the transmission distance of a packet, and prevent the packet from traversing a longer path introduced by a wormhole. First, when employing a temporal leash, a sender adds this leash and the sending time to a packet and sends this packet. Second, a receiver obtains the packet and derives the leash. Last, the receiver uses the receiving time to compare it with the leash and sending time for recognizing the wormhole attack. Similarly, when using a geographical leash, a sender generates a packet including this leash, the sending time and its location information, and then sends the packet. With the received packet, a receiver extracts the leash, the sending time and the location information of the sender, and compares the extracted message with the receiving time and its location for detecting the wormhole attack. When a receiver collects security-related data (i.e., packet leash, sending time, and location information) from a sender, the trustworthiness (i.e., the quality) of collected security-related data was considered since the sender restricts the transmission distance of a packet and should be trusted and authors introduced the error. In addition, the integrity, the authentication and the non-repudiation of original data are achieved, because authors employed a key to generate a message authentication code. The real time is achieved since an end-to-end delay extracted due to the sending time and the receiving time is directly used to detect the wormhole attack.

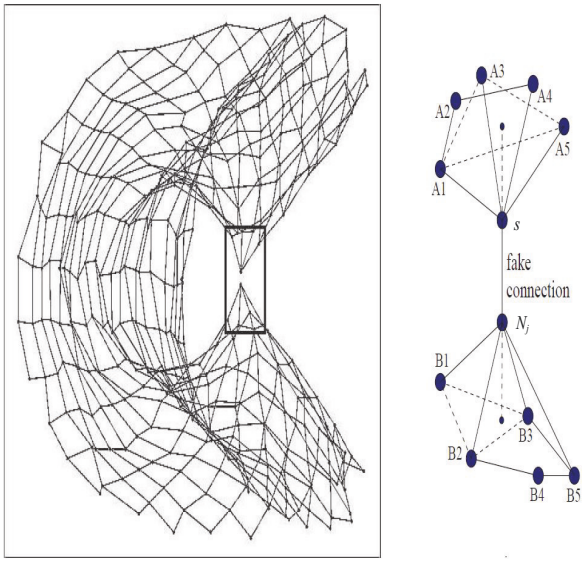


Figure 2: A two-end torch structure introduced by a wormhole [17]

Nevertheless, the confidentiality, the privacy and the stability were not considered, and authors did not consider the synchronization of generating the sending time and the location information. It is noted that the collected data are original data, and thus the trustworthiness of alarm data and the non-repudiation of alarm data are not suitable for evaluating the above collection due to non-existent alarm data.

Multi-Dimensional Scaling (MDS) [41, 42] can help visualize a network by employing the distance of any pair of neighbors and thus discover a torch structure to determine a fake link (i.e., wormhole link) [17]. First, a node measures the distance to its neighbor based on a RSS value [43], and sends a value of this distance and its neighbor list to a controller via an established route or flooding. Second, the controller computes the average of the distances reported by any pair of neighbors. Third, based on the Dijkstra algorithm, the MDS and a smoothing algorithm, this controller constructs a visual network. Finally, the controller computes the wormhole indicator of a node s, $max\{counter\,of\,sN_j, j = 0, \ldots, k-1\}$, where N_j is a neighbor of s and the counter is the number of two-end torch structures formed by a link sN_j. In more detail, there are two plane sets constructed by the neighbors of s and N_j respectively, and a two-end torch structure is constructed as shown in Figure 2 if the angles between a link sN_j and the two planes, which are selected from the two sets respectively are both larger than $3\pi/8$. When a controller collects a distance and a neighbor list from flooding, the stability was considered since flooding improves the probability that these security-related data arrive at this controller. However, authors did not consider other requirements. Note that it is not suitable to employ the trustworthiness of alarm data and the non-repudiation of alarm data for evaluating the above collection since the distance and neighbor list can be regarded as original data and there are no alarm data.

Each node can detect the out-of-band wormhole attack by finding forbidden substructures in its local connectivity graph [44]. Each node first collects connectivity information coming from the upper layer protocols such as routing protocol. If there is an out-of-band wormhole link, two node sets, whose members are located in the neighborhoods of two endpoints of this link respectively consider each other as a neighbor. In the same set, two nodes might

not be neighbors, but have common and independent (i.e., non-neighbor) neighbors in the opposite set. Then, each node counts the common and independent neighbors. The out-of-band wormhole attack is identified if the number of the common and independent neighbors exceeds a threshold value determined according to the unit disk graph feature in a normal network graph. In the same scenario, after each node collected connectivity information, removing a small part of the neighborhood of an out-of-band wormhole link [45] leads to the case that the remaining neighborhood is disconnected and divided into two components.

In the Split Multi-path Routing (SMR) [46] protocol, any intermediate nodes are required not to consider the incoming link of a duplicate RREQ. Based on the SMR protocol, Song et al. proposed to detect the wormhole attack [47] by discovering a link that appears with an abnormally high frequency. First, in the route discovery phase of SMR protocol, a destination node determines all available routes from a source node by collecting RREQs within a fixed period. Second, the destination node computes the relative frequency of each link appearing in the discovered routes, denoted as $p_i = n_i/N$, where n_i is the number of the times that the i-th link appears in the discovered routes and N is the number of links of all the routes. Also, it obtains an expression $\varphi = (n^{max} - n^{2nd})/n^{max}$, where n^{max} is the maximum value in a set $\{n_i\}$ and n^{2nd} is the second biggest value in the set $\{n_i\}$. Last, a link is regarded as a wormhole if both the maximum value of p_i and φ are larger than threshold values, respectively.

Similarly, in WARP [48], a wormhole node can be recognized by its neighbors if its extreme competition capacity is shown in the route discovery phase of Ad hoc On-demand Distance Vector (AODV) protocol. In other words, a node is a wormhole if it owns a high route-building rate that exceeds a preset threshold value. An originator first sends a Route Reply Decision (RREP-DEC) along a routing path if it receives a RREP from this route. Then, a node counts the received RREPs from its next hop neighbor and the RREP-DECs that it sends to this neighbor. In addition, it computes the anomaly value change of the neighbor, i.e., $(the\,number\,of\,RREP - DECs)/(the\,number\,of\,RREPs + 1)$, which represents the possibility that the neighbor is among the nodes in link-disjoint multiple paths. As a result, the anomaly value change is compared with a threshold value for recognizing the wormhole attack.

3.1.3. Detection Mechanisms against Rushing Attack

Hazra and Setua proposed a context aware trusted AODV protocol [49] for detecting the rushing attack [50, 51, 52, 53] that an attacker launches by ignoring the delay of MAC or routing protocols and adopting a large transmission range (denoted as context-1 and context-2, respectively). It is worth noting that if a next hop node of a node does not adopt the large transmission range but the node, the transmission between these two nodes is unidirectional. In terms of context-1 detection, if a node (trustee) receives a RREQ in the route discovery phase of AODV protocol, it immediately broadcasts a response (RQres) packet for this received RREQ, and its next hop node (trustor) records the receiving time of this RQres. Then, the trustee broadcasts the received RREQ after processing, and the trustor still records the receiving time of this RREQ. Finally, the trustee is disbelieved by the trustor and obtains a low direct trust value if the delay between the receiving time of RQres and RREQ is less than standard time. With respect to context-2 detection, after a trustor received a RREQ from a trustee in the route discovery phase, it broadcasts a response (RQres) packet of this received RREQ. If the trustor cannot receive a response packet of the RQres from the trustee within specific time, the trustee obtains a low direct trust

value and is related to the context-2. In addition, a trustor can receive indirect trust values about a trustee from recommenders. Therefore, the trustor can obtain the final trust value of a trustee by comprehensively considering its historical trust value and direct trust value about this trustee, and the indirect trust values. Ultimately, the trustor compares the final trust value with the threshold of 0.5 for determining whether it should forward the RREQ from the trustee. When a trustor records the receiving time of RREQ, RQres, and response packet of RQres, the trustworthiness of receiving time is achieved since a trustor should be trusted by itself and can employ standard time and specific time to mitigate the time error introduced by objective reasons such as hardware failure. Moreover, the trustworthiness of indirect trust value was also considered since the trustor contributes all received indirect trust values from different recommenders to a final trust value and the majority of recommenders should be trusted. The collection also achieves the real time requirement due to the specific timeout. However, other requirements were not considered by the authors.

3.1.4. Detection Mechanisms against Black Hole Attack

In [54], Su proposed a detection mechanism based on the principle that in the route discovery phase of modified AODV protocol an intermediate node only broadcasts a received RREQ but does not reply to this RREQ if it is not a destination node. After monitoring that a non-destination intermediate node fails to broadcast a received RREQ for a specific route but gives a reply, an IDS node increases the suspicion value of this node by 1. Therefore, the IDS node considers the intermediate node malicious when this value exceeds a threshold value.

By using the fact that black hole nodes forward either no RREQs (single black hole) or few RREQs (cooperative black hole) in the route discovery phase of AODV protocol, Imran et al. proposed a Detection and Prevention System (DPS) [55]. Specifically, a DPS node can monitor whether its neighbor forwards a RREQ. Then, the DPS node counts RREQs forwarded by the monitored neighbor. If the DPS node finds that the neighbor is with a low rate of broadcasting RREQs, it increments the suspicion value of this neighbor. Last, the neighbor is deemed malicious when the value exceeds some threshold value. However, the authors considered no available requirement when designing the collection of the number of RREQs. Note that the number of RREQs belongs to original data, the trustworthiness of alarm data and the non-repudiation of alarm data are not taken into consideration due to the non-existence of alarm data. Furthermore, the synchronization should not be considered since a DPS node constantly collects the number of RREQs forwarded by a monitored neighbor for each time of detection.

To detect the black hole attack in AODV protocol, Biswas et al. allowed a source node to monitor whether a discovered route feedbacks an ACK for a sent data packet [56]. First, a source node discovers some routes to a destination node in the route discovery phase. Second, by employing the randomly assigned rank, the velocity and the battery power of nodes in the discovered routes, the source node computes the trust value of these routes, and then selects a route with the largest trust value for data packet transmission. Third, after sending a data packet during the data packet transmission phase, the source node waits for an ACK coming from the destination node. Fourth, once receiving the data packet, the destination node sends an ACK to the source node along the selected route. Finally, if the source node fails to receive the ACK within specific time, it reduces the trust value of the route, and a new route with the largest trust value is selected from the available routes for data packet transmission.

Yu et al. also proposed a detection mechanism [57] by employing each node to monitor the packet forwarding

behavior of its neighbors. A node considers its neighbor suspicious if it finds that this neighbor fails to route any data packets successfully from or through it. Additionally, when the receiving/transmission ratio of the suspicious neighbor is high, the node sends a RREQ to a cooperative node selected from the neighbors of the neighbor, and a route through the neighbor thus can be discovered. Then, the node sends a check packet to the cooperative node and asks it whether the check packet is received or not. If the result is negative, the node requires all the neighbors of the neighbor to cooperatively identify its malevolence.

Similarly, Djenouri and Badache presented a detection mechanism [58] by considering the packet forwarding behavior of each node. Specifically, when transmitting a data packet in the data packet transmission phase, a node (i.e., monitor) randomly requests its 2-hop node en route for an ACK. If an ACK is not received within standard time, the node considers the presence of an intermediate node (i.e., its monitored 1-hop node) dropping data packets. Then, by employing the Bayesian approach, the node increases the reputation of the monitored node without the case of dropping data packets but otherwise reduces it. Also, the node makes a judgement after collecting some observations (i.e., dropping data packets and no dropping data packets). Finally, the result of this judgement is compared to a threshold value so as to determine the existence of the black hole attack. When a monitor collects an ACK from a 2-hop node en route, the integrity, the authentication, and the non-repudiation of the collected ACK are ensured due to the adoption of the encryption and decryption on a random number. Moreover, the real time is ensured due to standard time. The stability was also taken into account since the packet forwarding behavior of a monitored node is directly used for the detection of black hole attack. However, other requirements were not considered. Obviously, the ACK does not belong to alarm data but original data, and thus the trustworthiness of alarm data and the non-repudiation of alarm data should not be taken into consideration. Additionally, the synchronization should not be discussed, because the monitor collects the ACK from the 2-hop node en route for observing the packet forwarding behavior of the monitored node.

In the TOpology Graph Based Anomaly Detection (TOGBAD) [59] developed for an Optimized Link State Routing (OLSR) protocol, a modified cluster-based anomaly detector [60, 61] is employed to construct a topology graph based on the routing and data packets of all nodes, and compares this topology graph with the number of neighbors reported by each node for recognizing the black hole attack. First, a detector collects routing and data packets from each node, and constructs a topology graph. Second, if a node receives a HELLO message, it extracts the number of neighbors of this message's originator, and sends the number to the detector. Finally, the detector compares the received number with a number derived from the topology graph. As a consequence, an alarm is triggered when a difference extracted from the comparison is greater than a threshold value.

In the route discovery phase of AODV protocol, a novel honeypot based detection and isolation approach [62] allows a detector called honeypot to broadcast spoofed RREQs to attract replies from malicious neighbors. Compared to an original RREQ, the spoofed one has two different fields: the identification of a non-existent node and Time-To-Live (TTL) = 1. To participate in a route, a black hole neighbor feedbacks the reply with the highest sequence number, and thus it can be recognized.

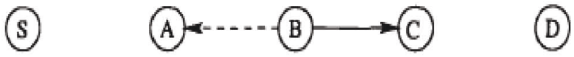


Figure 3: Watchdog technique illustration [68]

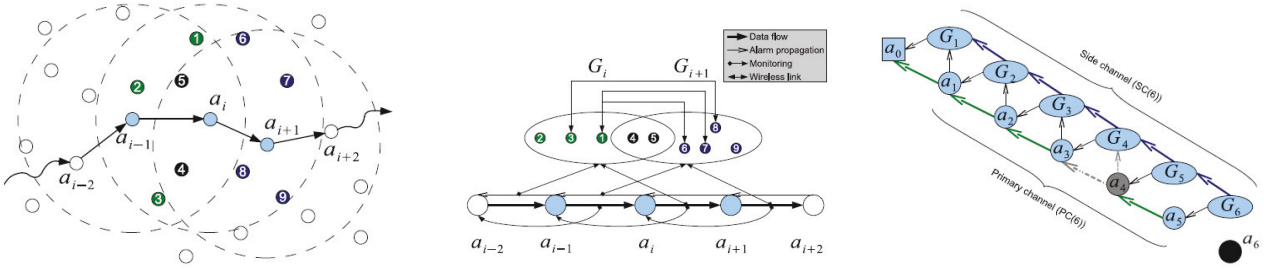
3.1.5. Detection Mechanisms against Grey Hole Attack

To detect the grey hole attack in an AODV protocol, Sen et al. considered the packet forwarding behavior of nodes [63]. If a node (detector) has not communicated with its neighbor recently, it further detects this neighbor. In the further detection process, the detector first discovers a route passing the neighbor to some reliable neighbor of this neighbor. Then, the detector sends a probe packet along the established route to the reliable node, and asks this reliable node whether it has received the probe packet. If the reliable node fails to receive the probe packet, the detector requires each neighbor of the suspicious neighbor to send three further probe packets for detecting the suspicious neighbor. The suspicious neighbor is thereby deemed malicious when the three further probe packets are dropped.

Gao and Chen proposed a detection mechanism [64] by using a checkup algorithm. First, in the data packet transmission phase of DSR protocol, each node is required to generate evidence [65] on a forwarded packet. Second, if a source node suspects the presence of the behavior of dropping data packets (e.g., it receives fewer packets from a destination node than that under a normal circumstance), it initiates a checkup algorithm to return the evidence. Last, the source node employs the returned evidence to trace a malicious node.

Gurung and Chauhan proposed the Mitigating Gray hole Attack Mechanism (MGAM) [66] by considering the packet forwarding behavior of each node. In the data packet transmission phase of AODV protocol, several Grey hole-Intrusion Detection System (G-IDS) nodes are employed to overhear the transmissions of their neighbors and compute the difference of the number of data packets that each neighbor receives and forwards respectively. Therefore, a G-IDS node compares the difference with a threshold value for recognizing the grey hole attack. Nevertheless, the above presented mechanism was not considered to meet the requirements as specified in Section 2 for the collection of the number of data packets. Note that the number of data packets belongs to original data and thus the trustworthiness of alarm data and the non-repudiation of alarm data are not suitable for evaluating the collection. In addition, the synchronization should not be considered since a G-IDS node continuously collects the number of data packets from an individual neighbor for each time of detection.

In the route discovery phase of AODV protocol, a trust model [67] can evaluate the trust value of a route based on traffic and select a route with a high trust value for data packet transmission, which helps to detect and prevent the grey hole attack. First, each node (i.e., monitor) monitors the traffic of its neighbor and periodically computes and broadcasts the trust value of it on the neighbor. Second, each common neighbor of the monitor and the monitored neighbor also derives and broadcasts the trust value of it on the monitored neighbor at a regular interval. Third, based on the received trust values, a source node can compute the trust value of a route including the monitors and monitored nodes and choose the route with a high trust value for transmitting data packets. However, any proposed requirements were not considered in traffic collection.



(a) Route layout (b) Monitoring relation (c) Alarm channel

Figure 4: SCM illustration [72]

3.1.6. Detection Mechanisms against Packet Dropping Attack

In [68], Marti et al. proposed a watchdog technique by considering that each node en route observes whether its neighbors forward received packets. First, each node en route serves as a watchdog to monitor the transmissions of its neighbors. As depicted in Figure 3, A saves the copy of a received packet from S and forwards it to B. When B forwards the received packet to C, A also hears the packet from B since A is located at the transmission range of B. Second, if A fails to hear the packet from B within fixed time, it decreases the confidence level of B by 0.05. Otherwise, A increases the confidence level of B by 0.01. Last, when the confidence level of B is equal to 0, an alarm is launched to discover a new route without including B.

A TWOACK technique [69] was proposed based on the fact that a node en route should receive a TWOACK packet from its 2-hop node after sending this 2-hop node a data packet. In order to reduce communication burdens, a mechanism named S-TWOACK allows the 2-hop node to send a TWOACK packet for acknowledging received multiple data packets. If the node fails to receive the TWOACK packet within specific time, the link between its 1-hop node and the 2-hop node is suspected, and the node increments the number of misbehavior instances by 1. Finally, the number of misbehavior instances is compared to a threshold value for determining the packet dropping attack.

Based on the principle that a source node receives an ACK from a destination or sink node after sending a data packet, Pu and Lim proposed a detection mechanism [70] against the packet dropping attack. First, in the data packet transmission phase, a source node randomly chooses a node en route as a checkpoint node, and generates and sends a data packet that includes a random number. Second, a node can know whether it is the checkpoint node by using the one-way function and the map function [71]. Third, the checkpoint node and a destination node send the source node ACKs respectively after receiving a data packet. Fourth, if a node en route fails to receive an ACK or alarm packet from its downstream node within a certain time period, it sends a generated alarm packet to the source node and retransmits the data packet. Another alarm packet is generated and sent if the node still fails to obtain an ACK or alarm packet. Ultimately, the source node compares the number of the received alarm packets to a threshold value. ACK or alarm packet collection ensures the real time since it adopts a certain time period. Other presented requirements, however, were not considered during the ACK or alarm packet collection. One notes that the synchronization should not be taken into consideration since a source node constantly collects the ACK from checkpoint and destination nodes or the number of the alarm packets for detection.

Using the principle that observers monitor the packet forwarding behavior of their neighbors, a Side Channel Monitoring (SCM) scheme [72] was proposed to detect the packet dropping attack. In addition to a primary channel (i.e., a selected route), a side channel is constructed by some neighbors of each node en route, and can transfer alarm packets generated by these neighbors called observers. Assume nodes $a_0, a_1, \ldots, a_{k+1}$ form a route, where a_0 and a_{k+1} are a source node and a destination node respectively. As depicted in Figure 4(a), if a_{i-1} has sent a_i a packet, a set of observers denoted as $G_i = \{4, 5, 6, 7, 8, 9\}$ monitors whether a_i forwards the received packet to a_{i+1}. Figure 4(b) represents a monitoring relation. In Figure 4(c), when G_i finds that a_i fails to forward the packet towards a_{i+1} within specific time, it sends an alarm packet to a_0 in reverse to the direction of relaying packets. Last, a_0 accumulates the received alarm packets for a_i, and compares the number with a threshold value.

Also monitoring the misbehaviors (i.e., packet dropping attack) of forwarders en route, Lee and Choi proposed a neighbor watch system [73]. In the data packet transmission phase, when the neighbor of a monitored node en route collects data packets to check the behavior of the node, the integrity, the authentication and the non-repudiation of original data are achieved due to the adoption of a shared key and a message authentication code. Other presented requirements however were not taken into consideration. It is noted that the trustworthiness and non-repudiation of alarm data should not be considered since the collected data are not alarm data but original data. The synchronization also should not be discussed since the neighbor collects the data packets from the single node instead of more nodes for each time of detection.

After a node collected the traffic of its neighbors, it can employ the flow conservation [74, 75, 76, 77] for recognizing the packet dropping attack. Anjum and Talpade proposed a practical approach [78] by analyzing and correlating reported statistics on packets originated, received and forwarded. First, each node en route counts packets originated, received and forwarded. Second, it reports the obtained number to a coordinator node periodically. Last, the coordinator node can analyze and correlate statistics extracted from the reports for identifying a malicious node.

3.1.7. Detection Mechanisms against Sleep Deprivation Attack

In the route discovery phase, an attacker could send a victim a lot of RREQs, RREPs and (Route Errors) RERRs to drain the battery power of the victim. Considering the detection of this kind of sleep deprivation attack, Sarkar and Roy allowed a cluster head node to count control packets (i.e., RREQ, RREP and RERR) from its cluster members [79]. In detail, when a source node in a cluster sends control packets to a destination node in another cluster via the cluster head nodes of both the clusters, the cluster head node to whom the source node belongs records the number of the control packets in a certain period. Therefore, the source node is deemed malicious if the number exceeds a threshold value. In the route discovery phase of AODV protocol, when an attacker broadcasts excessive RREQs, the route tables of nodes are overflowed. To overcome this flaw, Yi et al. proposed a detection mechanism [80] based on that each node counts RREQs from its neighbors directly. Specifically, a node first collects the number of RREQs from its neighbor. Then, the node computes a priority rule for the neighbor, which is inversely proportional to the collected number. If the neighbor broadcasts excessive RREQs at a regular interval, the priority rule for this neighbor is very low. Last, the node refuses future RREQs from the neighbor if the case that the priority rule is too low appears frequently. Nevertheless, any presented requirements were not considered

in [79, 80] when the number of these control packets is collected. Similarly, the trustworthiness of alarm data and the non-repudiation of alarm data should not be considered since one deals with original data and there are no alarm data. Moreover, a node collects the number of the control packets from its single neighbor for each time of detection continuously, and thus it is not necessary to take the synchronization into consideration.

To launch the sleep deprivation attack in the route discovery phase of AODV protocol, an attacker broadcasts a RREQ, which includes the IP address of a non-existent destination node. To detect this kind of attack, Chaudhary et al. proposed a distributed and cooperative intrusion detection system [81] based on an adaptive neuro-fuzzy inference system and a subtractive clustering method, which needs to collect some necessary features (i.e., the hop counts of active nodes, the serial number of RREQs, RREPs, data packets and neighbors, and consumed batteries).

In a study [82], Martin et al. proposed a detection mechanism by using the fact that the power of a node in an idle state is lower compared with the power of this node under the sleep deprivation attack. A node measures the average power usage of a node in an idle state and under the sleep deprivation attack respectively. Therefore, the difference of the average power usage can reflect the impact of the sleep deprivation attack and is thereby used to detect this attack.

3.1.8. Detection Mechanisms against Sybil Attack

In [83], Piro et al. proposed a detection mechanism against the Sybil attack based on the fact that multiple Sybil identities that a mobile node adopts should move consistently (i.e., these Sybil identities will be overheard or not be overheard simultaneously). Specifically, an observer first counts intervals in which two nodes are observed simultaneously and the number of periods in which either of the nodes is observed. Second, the observer computes the affinity value between both the nodes by using the collected number. Third, the observer constructs a graph, where a vertex and an indirect edge represent the identity of a node and the computed affinity value of each pair of nodes respectively. Last, a depth-first search running over each vertex is employed to discover the largest connected component (i.e., an attacker). During the collection of the number of intervals, the synchronization is achieved since the interval represents the simultaneous appearance of two nodes or the appearance of one of both nodes. However, other presented requirements were not considered reasonably. It is noted that the evaluation of the collection does not employ the trustworthiness of alarm data and the non-repudiation of alarm data, because the number of intervals does not belong to alarm data but original data.

Also making use of the mobility of nodes, a detection mechanism [84] was proposed against the Sybil attack. Similarly, all Sybil identities belong to a malicious node, and thus a watchdog node discovers all of the Sybil identities simultaneously when the malicious node moves to the neighborhood of the watchdog node. Specifically, a watchdog node owns a unique label, and it assigns its label to a discovered identity after ensuring that other watchdog nodes did not discover this identity. In addition, the watchdog node computes the number of the identities that have the same label. Last, if the number exceeds a threshold value, the identities are considered to belong to a Sybil attacker.

Ssu et al. proposed a detection mechanism [85] based on the fact that the probability that two nodes have exactly identical neighbors is low in a high density network. First, a node broadcasts a request message. Second, after receiving the message, a neighbor of the node broadcasts a message. Third, each node who hears the message

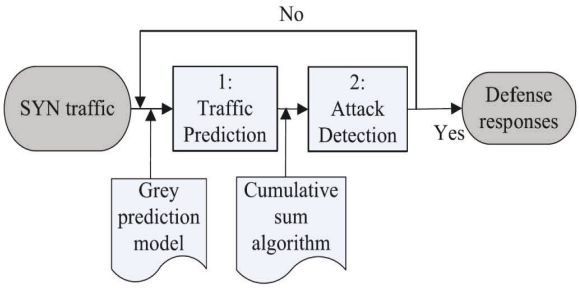


Figure 5: Detection mechanism based on traffic prediction [88]

replies to the neighbor. Fourth, the node records the identities of nodes whose replies it can hear. Therefore, the node can have knowledge about the common neighbors of its each neighbor and itself. Last, the node computes the number of times that an identity appears in the common neighbors, and thus a malicious node is determined if the number is greater than a threshold value.

A detection mechanism [86] was proposed based on the principle that a node finds that the first RSS of a legitimate node is low enough when the latter enters the radio range of the former. In detail, a receiver measures RRS from other senders continuously. If the receiver finds that a new identity appears abruptly with high RRS, it deduces that an attacker who is already located at its radio range creates the identity. Unfortunately, when a receiver collects RSS from a sender, any proposed requirements were not taken into account. Note that the trustworthiness of alarm data and the non-repudiation of alarm data should not be considered since the receiver does not collect alarm data but original data (i.e., RSS). Also, the synchronization is not suitable for the evaluation of the RSS collection, because the receiver measures the RRS from its single neighbor (i.e., the sender) instead of more neighbors for each time of detection constantly.

3.1.9. Detection Mechanisms against SYN Flooding Attack

Geetha and Sreenath proposed a detection mechanism [87] by directly analyzing the number of half open connection states. At first, in the TCP of AODV protocol, a Multimedia Server (MS) collects the number of received SYNs, sent ACKs and final acknowledgements. Then, the MS counts half open connection states. If the number of the half open connection states is greater than a threshold value, the MS determines a malicious node.

By using a grey prediction model and a cumulative sum (CUSUM) algorithm, Wang et al. proposed a detection mechanism [88] against the SYN flooding attack. First, a MS collects the number of received SYNs from clients in the TCP. Second, as shown in Figure 5, the MS adopts a grey prediction model to predict SYN traffic flows with the collected number. Third, based on these predicted SYN traffic flows, a CUSUM algorithm is employed to check current SYN traffic flows for identifying the SYN flooding attack. However, an arbitrary presented requirement was not considered properly during the collection of the number of SYNs. We should not take into account the trustworthiness of alarm data and the non-repudiation of alarm data since we consider that the number of SYNs does not belong to alarm data but original data. Moreover, a MS continuously records the number of the received SYNs for detecting the attack on itself, and thus it is not necessary to consider the synchronization.

A detection mechanism [89] was proposed based on the fact that some sequences of SYN, SYN-ACK and RTS

possibly imply the DoS attack. At first, a last mile router observes TCP handshakes (i.e., the sequence of SYN, SYN-ACK, and RTS) between a MS and a client. Subsequently, relying on unusual TCP handshakes, the router computes information entropy. Last, there is the SYN flooding attack if the derived information entropy is less than a threshold value. Also, a Repeated Threshold Violation (RTV) [90] can be used to avoid a false alarm. However, this work does not discuss the proposed requirements during the collection of the sequence of SYN, SYN-ACK and RTS. Since the collected data are original data, there are no alarm data and a last mile router collects the sequence for a MS and a client for each time of detection, the trustworthiness of alarm data, the non-repudiation of alarm data and the synchronization should not be discussed.

Korczynski et al. proposed a detection mechanism [91] based on the principle that a connection is established successfully if all packets from a MS include an ACK flag set on and a SYN flag set off. After a broader router sampled an arbitrary outgoing SYN, it first checks related time. Then, based on the result, the router decides to either allow the sampled SYN to pass or increase a request counter R_{dst}. Since an arbitrary incoming packet from a MS with a set ACK flag and a disabled SYN flag implies a legitimate connection, the router is able to decrease R_{dst}. As a consequence, $R_{dst} > R_{dst}^{max}$ means that there are a large number of connections to a destination address.

3.1.10. Detection Mechanisms against Man-In-The-Middle Attack

The session hijacking attack is a kind of the man-in-the-middle attack. Long and Sikdar proposed detection mechanisms [92, 93] against the session hijacking attack by employing the abrupt change of RSS caused by this attack. At first, a receiver measures RSS from a sender. Afterwards, the receiver employs a step function to describe the change of the RSS. Last, an optimal filter is constructed to identify the session hijacking attack. Nevertheless, any presented requirements were not considered when a receiver collects RSS from a sender. Also, since the RSS belongs to original data and the receiver measures the RSS from the single sender instead of more senders for each time of detection, the trustworthiness of alarm data, the non-repudiation of alarm data and the synchronization should not be discussed.

In the data packet transmission phase, positive acknowledgement in MAC protocol requires that a receiver must send an ACK within limited time after receiving a packet from a sender. To ensure that the sender receives the ACK within the expiration of time, an attacker itself sends the sender the ACK whenever it receives the ACK from the receiver. Glass et al. proposed a detection mechanism [94] against this attack by using a specific packet that should not be acknowledged. Specifically, a sender and a receiver secretly agree that designed packets should not be acknowledged on their first transmission. Then, if the receiver obtains a specific packet, it should not feedback an ACK to the sender. Since an attacker has no idea about the specific packet, it will send the ACK to the sender, and thus is recognized. During ACK collection, the real time is achieved due to specific expiration time. However, the collection fails to ensure other proposed requirements. The trustworthiness of alarm data, the non-repudiation of alarm data and the synchronization should not be taken into account, because an ACK belongs to original data and a sender does not collect the ACK from at least two receivers but a single receiver for each time of detection.

Aziz and Hamilton proposed a detection mechanism [95] against the man-in-the-middle attack based on that a distance-bounding protocol [96] requires a packet to reach its destination timely. A receiver first records the time at which it receives a packet. Then, the receiver employs a precise timing based static analysis to derive name

substitutions as the result of communication. Last, the result is able to be employed to define a name integrity property and the notion of the man-in-the-middle attack.

After obtaining cross-information from distinct sources by using an automated and distributed method, a man-in-the-middle distributed assessment system [97] classifies certificates into a trusted class and a non-trusted class when the certificates pass through a specific network. For a query assessment, e.g., whether a Transport Layer Security (TLS) handshake is trustworthy or not, a detector, who has an embedded Bayesian network issues the assessment directly extracted from network evaluation. If a node requires obtaining the assessment value about the trustworthiness of a specific TLS exchange, its own assessment and the assessments, which are collected from some randomly chosen nodes are integrated. Therefore, the node employs the integrated result to decide whether the TLS exchange is accepted or not.

3.1.11. Detection Mechanisms against Worm Attack

Considering the worm propagation over the TCP transport, Kim et al. proposed a detection mechanism [98] based on a signature including IP protocol number, destination port number and byte sequence. The pattern to generate a signature is called an autograph, which is based on single substring match. If a node finds that the payload in received network flows matches the byte sequence for the same IP protocol, it considers the payload as a worm. When a node collects the feature of packets (i.e., IP protocol number, destination port number, and byte sequence), none of proposed requirements were considered. We should not discuss the trustworthiness of alarm data, the non-repudiation of alarm data and the synchronization, because the feature does not belong to alarm data but original data, and a node collects the feature from a single data provider instead of more data providers for each time of detection.

A polygraph approach [99] employs multiple substrings to generate signatures for detecting a worm that attempts to modify the sequence of byte flows. In the polygraph, a token-subsequence signature develops a Simplified Regular Express (SRE) [100]. On the basis of multiple sequence alignment, Tang et al. proposed a detection mechanism [101], which maintains the distance of invariant contents and the wildcard string alignment. Also, a signature based on the SRE is able to represent the distance information for the invariant contents, which assists to detect the polymorphic worm.

Based on the relation of SRE signatures, a new Network-based Signature Generation (NSG) system called PolyTree [102] was proposed against the polymorphic worm attack. A node first derives signatures from worm samples. Then, it constructs a tree based on the obtained signatures. The PolyTree can generate accurate signatures which reflect the relations for multiple worms.

Wang et al. proposed a detection mechanism [103] which generates length-based signatures for detecting the worms of buffer overflows. The main idea is that to adopt the vulnerability of buffer overflows, an attacker keeps the length of protocol fields long enough. Concretely, a vulnerability happens if some protocol message is mapped into existing vulnerable buffers. When an attacker sends a lot of messages to the designed field of a protocol for buffer overflows, this field is much longer than that of a normal request, which thereby can be employed to detect worms. Similarly, any proposed requirements were not taken into consideration during the collection of the specific field length of a protocol. The trustworthiness of alarm data and the non-repudiation of alarm data should not be

Table 2: Comparison of Security-Related Data Collection Methods in Single Point Detection Mechanisms

Detected Attack	Reference	Collection Time	Data Provider	Data Forwarder	Data Collector	Collection Mode	Collected Data Type
Jamming attack	[21]	N	Message sender	NE	Message receiver	N	Packet, RSS
	[24]	N	Message sender	NE	Message receiver	N	PDR
	[28, 29]	Data packet transmission phase	Destination node	N	Each node	MAC protocol	The number of received CTSs
Wormhole attack	[39, 40]	N	Message sender	N	Message receiver	N	Packet leash, sending time, location information
	[17]	N	Each node	Nodes en route or each node	Controller	Established route or flooding	Distance, neighbor list
Rushing attack	[49]	Route discovery phase	Trustee, recommender	N	Trustor	AODV protocol	Receiving time of RREQ, RQres, and response packet of RQres, indirect trust value
Black hole attack	[55]	Route discovery phase	The neighbor of DPS node	NE	DPS node	AODV protocol	The number of RREQs
	[58]	Data packet transmission phase	2-hop node en route from monitor	Monitored 1-hop node en route from monitor	Monitor en route	N	ACK
Grey hole attack	[66]	Data packet transmission phase	The neighbor of G-IDS node	NE	G-IDS node	AODV protocol	The number of data packets
	[67]	Route discovery phase	Monitored neighbor	NE	Monitor	AODV protocol	Traffic, trust value
Packet dropping attack	[70]	Data packet transmission phase	Nodes en route	Upstream nodes of data provider	Source node	N	ACK, alarm packet
	[73]	Data packet transmission phase	Nodes en route	NE	The neighbor of the nodes en route	N	Data packet
Sleep deprivation attack	[79]	Route discovery phase	Source node	NE	Cluster head node	N	The number of RREQs, RREPs and RERRs
	[80]	Route discovery phase	Monitored node	NE	The neighbor of monitored node	AODV protocol	The number of RREQs
Sybil attack	[83]	N	The neighbor of observer	NE	Observer	N	The number of intervals
	[86]	N	Monitored node	NE	The neighbor of monitored node	N	RSS
SYN flooding attack	[88]	N	Client	NE	MS	TCP	The number of received SYNs
	[89]	N	Client, MS	N	Last mile router	TCP	The sequence of SYN, SYN-ACK and RTS
Man-in-the-middle attack	[92, 93]	N	Message sender	NE	Message receiver	N	RSS
	[94]	Data packet transmission phase	Message receiver	N	Message sender	MAC protocol	ACK
Worm attack	[98]	N	N	N	Each node	N	IP protocol number, destination port number, byte sequence
	[103]	N	N	N	Each node	N	Length of designed field of protocol

N: not specified or supported; NE: non-existent.

Table 3: Evaluation of Security-Related Data Collection Methods in Single Point Detection Mechanisms

Reference	TSRD		Co	Pr			In	Au	NR		RT	St	Sy
	TO	TA		IPr	LPr	RPr			NRO	NRA			
[21]	N	NS	Y	N	N	N	Y	Y	Y	NS	N	N	Y
[24]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[28, 29]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[39, 40]	Y	NS	N	N	N	N	Y	Y	Y	NS	Y	N	N
[17]	N	NS	N	N	N	N	N	N	N	NS	N	Y	N
[49]	Y	Y	N	N	N	N	N	N	N	N	Y	N	N
[55]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[58]	N	NS	N	N	N	N	Y	Y	Y	NS	Y	Y	NS
[66]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[67]	N	N	N	N	N	N	N	N	N	N	N	N	N
[70]	N	N	N	N	N	N	N	N	N	N	Y	N	NS
[73]	N	NS	N	N	N	N	Y	Y	Y	NS	N	N	NS
[79]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[80]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[83]	N	NS	N	N	N	N	N	N	N	NS	N	N	Y
[86]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[88]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[89]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[92, 93]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[94]	N	NS	N	N	N	N	N	N	N	NS	Y	N	NS
[98]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[103]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS

NS: not suitable for evaluation; Y: supported or considered.

taken into account since the specific field length does not belong to alarm data but original data. In addition, we should not discuss the synchronization since each node measures the specific field length of a protocol and does not collect the security-related data from at least two nodes for each time of detection.

3.2. Intrusion Detection Mechanisms

An IDS is able to recognize some attacks at a time. Due to used detection methods, the IDS can be mainly categorized into five classes [4]: Anomaly-Based Intrusion Detection System (ABIDS), Knowledge-Based Intrusion Detection System (KBIDS), Specification-Based Intrusion Detection System, Hybrid Intrusion Detection System (HIDS), and Other Intrusion Detection System (OIDS), which are introduced in Table 4. We review some existing IDSs, and then choose representative IDSs for comparing and evaluating involved security-related data collection methods in Table 5 and Table 6, respectively.

3.2.1. Anomaly-Based Intrusion Detection Systems

Based on the modification of Markov chain classifier, Sun et al. proposed an ABIDS [104] against the disruption attack that forged RREPs cause. In the route discovery phase of DSR protocol, a node collects following features as the input of the IDS, namely the change ratio of route entries and the change ratio of the number of hops.

In [105], Sun et al. proposed to detect local intrusions by employing the Markov chain and the Hotelling's T^2 test. The authors claimed that the mean speed of nodes is not able to represent the dynamics of MANETs and thus

Table 4: Intrusion Detection Systems

IDS	Description
ABIDS	The ABIDS derives a model (profile) according to acceptable activities and behaviors, and generates an alarm if monitored activities or behaviors tremendously deviate from this profile.
KBIDS	The KBIDS maintains the patterns of specific attacks and triggers an alarm if observed events match the patterns.
SBIDS	The SBIDS first extracts specifications that define correct operations (e.g., the operations of networks and protocols) with some constrains, and identifies an intrusion if monitored operations deviate from the specifications.
HIDS	The HIDS is the combination of ABIDS, KBIDS and SBIDS.
OIDS	The OIDS does not belong to the aforementioned IDS types.

it is not efficient to accommodate the impact of mobility. Therefore, a local IDS agent is allowed to periodically collect a change ratio of local links to adjust the effect of mobility.

In an ABIDS [106], Ye et al. used probabilistic techniques (e.g., Hotelling's T^2 test and decision tree) to investigate the frequency and the ordering property of audit data from various sources for recognizing intrusions. A node having this detection system collects audit data from Sun SPARC workstations and the MIT Lab to evaluate its performance.

Nadeem and Howarth proposed an Adaptive Intrusion Detection and Prevention (AIDP) system [107] against the DoS attack caused by the vulnerability of the route discovery of reactive routing protocols. Specifically, a cluster head node collects the number of RREQs received by each cluster node, and then takes the number as input. In the proposed system, the chi-square goodness of fit test [108] helps identify the presence of intrusions, and a control chart adopted in statistical process control [109] assists to recognize malicious nodes.

Zhang and Lee proposed an ABIDS [110] against various attacks at the network layer. In an architecture, each node acting as an IDS agent collects the Percentage of Changed Routes (PCR) and the Percentage of the Change in the Hop counts of all routes (PCH) from various sources, and then employs the proposed ABIDS to discover an abnormal change of routing tables for detecting various intrusions at the network layer. When an IDS agent collects PCR and PCH from a source, no presented requirements are ensured reasonably. Note that the trustworthiness of alarm data, the non-repudiation of alarm data and the synchronization are not suitable for evaluating the collection, since the PCR and the PCH are original data, there exist no alarm data and the IDS agent collects security-related data from a single source instead of more nodes for each time of detection. Subsequently in [111], Zhang et al. extended and simulated that architecture.

In [112], Sterne et al. constructed a cooperative IDS architecture against the packet dropping attack. At first, a leaf node acquires the data about nearby traffic flows either by comprehensively monitoring or from direct reports. Then, the leaf node sends the acquired data to a cluster head node. After receiving the data, the cluster head node executes the proposed IDS for detecting whether nodes drop a large number of packets deliberately or not. Similarly, when a cluster head node collects traffic flows from leaf nodes, the confidentiality, the integrity, the authentication and the non-repudiation of original data can be satisfied due to the adoption of cryptographic techniques. In addition, the proposed IDS is to detect the packet dropping attack and thereby the stability was considered. Unfortunately, other requirements are not met. It is worth noting that the trustworthiness of alarm data and the non-repudiation of alarm data should not be discussed since the collected data are original data and

no alarm data exist.

3.2.2. Knowledge-Based Intrusion Detection Systems

State transition analysis [113] can help to maintain well-known attacks for instance as a series of states. By employing the state transition diagram, an intrusion can be modeled as the sequence of the state changes from a secure state to a compromised state. Based on the state transition analysis, Vgina et al. proposed a tool named AODVSTAT [114] for detecting the packet dropping attack and the spoofing attack in AODV protocol. In AODVSTAT, a node (i.e., an observer) either observes traffic flows from its neighbors or collects UPDATE messages originated from other observers. Afterwards, the node analyzes the collected data. During the collection of traffic flows and UPDATE messages, the authentication is achieved since the AODVSTAT can detect the spoofing attack. When a node collects the traffic flows, it checks whether its neighbor forwards packets within specific time. Therefore, the collection achieves the real time. Moreover, the AODVSTAT is able to recognize the packet dropping attack, and thus the stability was considered. However, other requirements were not discussed. We should not discuss the trustworthiness of alarm data and the non-repudiation of alarm data since the traffic flows and UPDATE messages are original data.

Smith proposed a KBIDS architecture [115] based on static databases. An IDS agent detects local intrusions by using collected local audit data. Additionally, the IDS agent is able to cooperate with other IDS agents for determining an intrusion. When an IDS agent collects audit data, an arbitrary presented requirement was not discussed.

3.2.3. Specification-Based Intrusion Detection Systems

By using the finite state automata that determines the correct behaviors (i.e., specifications) of nodes with some constrains, Tseng et al. proposed a SBIDS [116] for OLSR protocols against the DoS attack. A node monitors the behaviors of its neighbors by collecting Hello messages, Topology Control (TC) messages and local data. Then, the node compares the monitored behaviors with specifications generated by the finite state automata for identifying the DoS attack. During the collection of Hello messages, TC messages and local data, the integrity and the real time were considered since the proposed SBIDS is able to ensure the integrity of route tables in each node and a specified correct behavior involves a fixed time period for receiving messages. However, other requirements were not taken into consideration. Similarly, we should not discuss the trustworthiness of alarm data and the non-repudiation of alarm data, because the collected data are not alarm data. Moreover, Orset et al. proposed an extended finite state machine [117] through manually extracting some constrains from an IETF specification [118]. By employing a backward tracking-based method [119], a node compares the trace of receiving and sending messages with specifications generated by the extended finite state machine for recognizing an intrusion. In [120], Tseng et al. also specified the correct routing behaviors of nodes with a set of constrains. In other words, a generated specification specifies the valid flow of routing packets for preventing from manipulating the routing packets.

Lin et al. proposed a SBIDS [121] by using a method called Previous Two Forwarders (PTF). The PTF can ensure the integrity of routing packets for preventing a message field from being tampered. Figure 6 shows an example. After receiving a RREQ M_{SA} from S, A sends B a PTF (i.e., M_{AB} and M_{SA}). Therefore, the PTF is able to ensure the correctness of mutual information and non-mutual information, where the mutual information represents hop

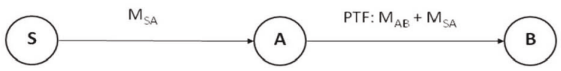


Figure 6: PTF illustration [121]

counts and the non-mutual information means other information such as sequence number, RREQ identity, and IP address. As a consequence, the PTF enforces the exchange of packets to comply with the specification of an AODV protocol. The collection of mutual and non-mutual information achieves the integrity, the authentication and the non-repudiation of original data, because the PTF aims to ensure the correctness of packet exchange. Note that we should not discuss the trustworthiness of alarm data, the non-repudiation of alarm data and the synchronization since we consider that the mutual and non-mutual information belongs to original data and a node collects a PTF from a previously single node instead of more nodes for each time of detection.

To detect the monitored abnormal behaviors of the run time of protocols, Stakhanova et al. proposed a SBIDS [122] based on the specification of protocols that can be extracted from network traffic flows and expressed as a graph. In the graph, a node represents a protocol configuration, and directed edges reflect the evolution from a configuration to another one.

3.2.4. Hybrid Intrusion Detection Systems

Routing protocols keep a routing communication minimal for ensuring efficiency, but thus minimal information caused is a barrier to a Routing Attack Detection System (RADS). In order to improve the accuracy of the detection against packet dropping attack, spoofing attack, and rushing attack, Joseph et al. proposed a Cross layer RADS (CRADS) [123]. The CRADS needs to collect evidence from various protocols at distinct layers (i.e., the network layer, the MAC layer and the physical layer) such that the information about routing behaviors increases. Based on the Support Vector Machine (SVM), the CRADS employs a non-linear detection technique to ensure the high efficiency of detection. Moreover, it adopts some data reduction approaches to reduce computation burdens introduced by a lot of extracted features and the use of SVM. Unfortunately, any presented requirements were not considered during the collection of evidence. It is not suitable to use the trustworthiness of alarm data and the non-repudiation of alarm data for evaluation since the evidence does not belong to alarm data.

By extending the AIDP system [107], Nadeem and Howarth proposed generalized intrusion detection and prevention systems [124, 125] for detecting various attacks at the network layer, which are constructed by combining the ABIDS with the KBIDS. In the route discovery phase of AODV protocol, a cluster head node collects a Network Characteristic Matrix (NCM) and a Derived Matrix (DM) from cluster nodes, where

$$\begin{cases} NCM = \{RREQ, RREP, RERR, TTL, src\_seq\,of\,RREQ, dest\_seq\,of\,RREQ, dest\_seq\,of\,RREP\} \\ DM = \{CPO(control\,packet\,overhead), PDR, CPD(the\,number\,of\,dropped\,packets)\} \end{cases}.$$

Also, the proposed systems use the collected data to recognize intrusions and malicious nodes in a testing phase, and generate profiles in a training phase. When a cluster head node collects the NCM and the DM, the real time and the stability were considered since both systems employ TTL, PDR and CPD for intrusion detection. However,

other requirements are not satisfied. The trustworthiness of alarm data and the non-repudiation of alarm data should not be taken into consideration, because the collected data are not alarm data.

3.2.5. Other Intrusion Detection Systems

There are some other intrusion detection systems whose types cannot be interpreted. Sanzgiri et al. proposed a hop-by-hop authentication scheme [126] for resisting attacks launched by external attackers, and more precisely the authors focused on spoofing, and modifying and fabricating packets in AODV and DSR protocols. The proposed scheme adopts a certification process between nodes to ensure the authentication, the integrity and the non-repudiation of messages, such that attacks introduced by spoofing and modifying and fabricating packets can be detected. When a node collects route information from another node (i.e., its neighbor) in a certification process, the integrity, the authentication and the non-repudiation of original data are achieved due to the goal of the proposed scheme. In addition, the real time aspect was considered since the node uses a timestamp for receiving route information. However, the collection fails to achieve or consider other requirements. Note that the trustworthiness of alarm data, the non-repudiation of alarm data and the synchronization should not be considered since the route information belongs to original data and the node does not collect this information from at least two nodes but a single node for each time of detection.

Yi et al. proposed a clustered intrusion detection system [127] for detecting the DoS attack and the routing loop attack in a DSR protocol. First, the proposed system randomly selects a node as a monitor. Second, the node monitors the behaviors of nodes in its cluster. Last, it detects intrusions locally and globally by checking the monitored behaviors with the help of a constructed finite state machine. When a monitor collects behaviors, any presented requirements however were not taken into consideration. Since these behaviors cannot be considered as alarm data but original data, the trustworthiness of alarm data and the non-repudiation of alarm data should not be discussed.

4. Open Issues and Future Research Directions

4.1. Open Issues

The completed comparison and evaluation shown in the above tables identify several open issues on security-related data collection in MANETs.

First, security-related data composition [14, 128] plays an important role in measuring the real-time security of MANETs and allowing the networks to react efficiently, but literatures lack investigation on security-related data composition. Based on Table 2 and Table 5, when we consider to integrate a number of detection mechanisms against main attacks for evaluating the security of MANETs, the integrated detection mechanisms need to collect a number of types of security-related data. In turn, real time of security-related data collection guarantees the performance of detection mechanisms, but this was seldom considered as can be seen from Table 3 and Table 6. A factor that impacts real time is the degree of data composition in the case that composed detection mechanisms need to collect various types of security-related data, since the transmission of duplicated data intuitively reduces the efficiency of security-related data collection. However, little work investigated security-related data composition for evaluating the real-time security of MANETs.

Table 5: Comparison of Security-Related Data Collection Methods in IDSs

IDS	Reference	Collection Time	Data Provider	Data Forwarder	Data Collector	Collection Mode	Collected Data Type	Detected Attack
ABIDS	[110]	N	Various sources	N	IDS agent	N	PCR, PCH	Various attacks at the network layer
	[112]	N	Leaf node	N	Cluster head node	N	Traffic flow	Packet dropping attack
KBIDS	[114]	N	Neighbor or other observers	N	Observer	AODV protocol	Traffic flow, UPDATE message	Packet dropping attack, spoofing attack
	[115]	N	N	N	IDS agent	N	Audit data	N
SBIDS	[116]	N	N	N	Each node	OLSR protocol	Hello message, TC message, local data	DoS attack
	[121]	N	Previous 1-hop node from monitor	N	Monitor	AODV protocol	Hop counts, sequence number, RREQ identity, IP address	The attacks caused by modification
HIDS	[123]	N	N	N	N	N	Evidence from various protocols at physical layer, MAC layer and network layer	Packet dropping attack, spoofing attack, rushing attack
	[124, 125]	Route discovery phase	Cluster node	N	Cluster head node	AODV protocol	NCM, DM	Various attacks at network layer
OIDS	[126]	N	The neighbor of data collector	N	Each node en route	AODV and DSR protocols	Route information	The attacks caused by spoofing, modification and fabrication
	[127]	N	Node in monitor's cluster	N	Monitor	DSR protocol	Node's behaviors	DoS attack, routing loop attack

Second, the trustworthiness of collected security-related data [8, 9, 10, 11] was rarely considered in existing data collection methods, but more attention should be paid to this issue. The result of further analysis might be false if the data with low trustworthiness are adopted for security measurement. Although some existing data collection methods allow a data provider to simply contribute its data to a detection result, the same weight is assigned to each data. For example, each data provider equally contributes its alarm data to a final detection result, and an attack is recognized if the number of alarms from different data providers exceeds a threshold value [70, 72]. Obviously, the data from an untrustworthy or low-trustworthy data provider should not be considered or fully considered in the detection.

Third, few existing security-related data collection methods satisfy the requirements on confidentiality, privacy, integrity, authentication and non-repudiation [12, 13, 14, 15]. Obviously, these requirements are crucial for achieving the trustworthy security measurement of MANETs. A general method to achieve the aforementioned properties is the cryptographic technique such as an encryption scheme and a hash function. However, it is not easy to achieve security-related data collection with confidentiality and non-repudiation in MANETs. From Table 3 and Table 6, we can see that some existing security-related data collection methods support non-repudiation, but most of them require security-related data receivers to release sensitive information to provide a proof such that data

Table 6: Evaluation of Security-Related Data Collection Methods in IDSs

Reference	TSRD		Co	Pr			In	Au	NR		RT	St	Sy
	TO	TA		IPr	LPr	RPr			NRO	NRA			
[110]	N	NS	N	N	N	N	N	N	N	NS	N	N	NS
[112]	N	NS	Y	N	N	N	Y	Y	Y	NS	N	Y	N
[114]	N	NS	N	N	N	N	N	Y	N	NS	Y	Y	N
[115]	N	N	N	N	N	N	N	N	N	N	N	N	N
[116]	N	NS	N	N	N	N	Y	N	N	NS	Y	N	N
[121]	N	NS	N	N	N	N	Y	Y	Y	NS	N	N	NS
[123]	N	NS	N	N	N	N	N	N	N	NS	N	N	N
[124, 125]	N	NS	N	N	N	N	N	N	N	NS	Y	Y	N
[126]	N	NS	N	N	N	N	Y	Y	Y	NS	Y	N	NS
[127]	N	NS	N	N	N	N	N	N	N	NS	N	N	N

confidentiality cannot be ensured. Therefore, it is also an open issue to enable security-related data collection with both the confidentiality and the non-repudiation. In addition, it is a challenge to minimize resource consumption caused by cryptographic techniques.

Fourth, stability [17, 18] was rarely taken into consideration in the existing work, which might cause that sufficient security-related data cannot be collected to ensure the accuracy of detection. On one hand, some existing security-related data collection methods use the method of flooding security-related data to ensure the reception of the data as much as possible, and thus provide high stability but waste resources. On the other hand, other methods allow a data provider to send security-related data to a data collector via an established route, and thereby save resources but suffer from low stability. However, the literature lacks an ideal solution that can balance stability and resource consumption (e.g., power) in a meaningful way.

Fifth, synchronization [19, 20] was seldom considered or achieved in existing security-related data collection methods. However, it is crucial to achieve the synchronization of various security-related data provided by different data providers, otherwise a detection result might deviate from the truth tremendously. Therefore, data synchronization becomes an essential requirement for the trustworthy security measurement of MANETs.

Although there are some existing schemes that could ensure many of the proposed requirements, a security-related data collection method that can satisfy all the proposed requirements is still lacking in MANETs. Therefore, new mechanisms should be explored in order to fulfill all the requirements towards trustworthy security-related data collection for MANET security measurement.

4.2. Future Research Directions

Through detailed analysis of the open issues, we suggest a number of research trends with regard to security-related data collection.

First, it is crucial to study an efficient, privacy-preserving and secure data aggregation method for ensuring the real-time security measurement of MANETs. In current security-related data collection methods, downstream data providers in a route not only forward security-related data from upstream data providers to a collector but also send their own security-related data to the same collector. Intuitively, the efficiency of security-related data collection is decreased due to data duplication and big data size. It is necessary to innovate a security-related

data collection method that composes or aggregates the data hop by hop in the route from a data provider to a data collector. At the same time, this aggregation process (in security-related data collection) is desired to ensure collection efficiency, privacy and all the proposed security properties (i.e., confidentiality, integrity, authentication, and non-repudiation). However, an efficient privacy-preserving and secure data aggregation for real-time security measurement is still missing in MANETs, although there exist many data aggregation schemes.

Second, data truth and quality discovery should be well studied for the purpose of the security measurement of MANETs. Only according to the collected security-related data, is it essential and hard to compute the truth of each security-related data. By processing given observation values (including the true data) from different data providers regarding the same object, existing truth discovery methods can find the true data and evaluate the trustworthiness of each data provider. The truth discovery might not be employed directly to derive the trustworthiness value of each collected security-related data, since data providers could generate the security-related data for different objects and these data might not contain the exact security-related data. Intuitively, distributed trust evaluation is able to help evaluate the trustworthiness of each collected data due to the distribution nature of MANETs. In addition, data clustering and classification involved in data mining might be employed to evaluate the trustworthiness of each collected security-related data. Simultaneously, we should enhance the efficiency of a designed scheme due to the constrained resource of MANETs.

Third, a future direction could be to design an incentive mechanism for encouraging data providers to honestly share their security-related data with the aim of improving the accuracy of security measurement. On one hand, we know that the high stability of security-related data collection guarantees the high accuracy of detection by enabling generated security-related data to reach a collector as far as possible. On the other hand, to ensure that enough security-related data can be collected for improving attack detection accuracy, we should also offer data providers some incentive for security-related data provision. There are many reasons for which data providers hesitate to share security-related data. For example, data providers might be overloaded, selfish, or in a situation of low-battery power. Therefore, it is a very interesting and significant issue to design an incentive mechanism for collecting sufficient and high-quality security-related data.

Fourth, data synchronization mechanisms should be further studied in order to achieve accurate security measurement. As we all know, it is hard to achieve the absolute synchronization in MANETs, but we should make efforts to reduce the deviation between a detection result and the true result, which could be introduced by non-synchronization. For example, the distance between a data provider and a data collector might be employed by the latter to estimate the time when the former generates security-related data. According to the results of estimation, the data collector can select the security-related data that satisfy some specific degree of synchronization. However, if the data collector chooses the security-related data with the high degree of synchronization for detection, there might be few selected data that are eligible, which instead increases the deviation between the detection result and the true one and decreases the accuracy of detection due to insufficient data for detection. Therefore, the trade-off between the degree of synchronization and deviation should be considered, and game theory might be an approach to solve this problem.

5. Conclusion

The detection mechanisms against main security attacks in MANETs require collecting security-related data. However, security threats also exist in security-related data collection. We identified a number of requirements towards trustworthy security-related data collection to combat these threats. Then, we thoroughly reviewed the existing attack detection mechanisms in MANETs and applied the proposed requirements as criteria to evaluate their performance in terms of trustworthy security-related data collection. Based on the survey and evaluation, we identified a number of open issues or challenges. As a consequence, related future research directions were formulated.

Acknowledgment

This work is sponsored by the National Key Research and Development Program of China (grant 2016YFB0800704), the NSFC (grants 61672410 and U1536202), the 111 project (grants B08038 and B16037), the Project Supported by Natural Science Basic Research Plan in Shaanxi Province of China (Program No. 2016ZDJC-06), and Academy of Finland (grant 308087).

References

[1] T.V.P. Sundararajan, S.M. Ramesh, R. Maheswar, et al., "Biologically inspired artificial intrusion detection system for detecting wormhole attack in MANET," Wireless Networks, vol. 20, no. 4, pp. 563-578, 2014.

[2] M.P. Arthur and K. Kannan, "Cross-layer based multiclass intrusion detection system for secure multicast communication of MANET in military networks," Wireless Networks, vol. 22, no. 3, pp. 1035-1059, 2016.

[3] G. Usha, M.R. Babu, and S.S. Kumar, "Dynamic anomaly detection using cross layer security in MANET," Computers and Electrical Engineering, vol. 59, pp. 231-241, 2017.

[4] A. Nadeem and M.P. Howarth, "A survey of MANET intrusion detection and prevention approaches for network layer attacks," IEEE Communications Surveys and Tutorials, vol. 15, no. 4, pp. 2027-2045, 2013.

[5] N. Deb, M. Chakraborty, and N. Chaki, "A state-of-the-art survey on IDS for mobile ad-hoc networks and wireless mesh networks," in 2011 1st International Conference on Parallel, Distributed Computing Technologies and Applications (PDCTA), pp. 169-179, 2011.

[6] E. Amiri, H. Keshavarz, H. Heidari, et al., "Intrusion detection systems in MANET: A review," Procedia - Social and Behavioral Sciences, vol. 129, no. 2, pp. 453-459, 2014.

[7] S. Jain and A. Khunteta, "Detection techniques of blackhole attack in mobile ad hoc network: A survey," in 2015 International Conference on Advanced Research in Computer Science Engineering and Technology (ICARCSET), pp. 1-5, ACM, 2015.

[8] M. Pouryazdan, B. Kantarci, T. Soyata, et al., "Anchor-assisted and vote-based trustworthiness assurance in smart city crowdsensing," IEEE Access, vol. 4, pp. 529-541, 2017.

[9] L. Liu, M. Esmalifalak, Q. Ding, et al., "Detecting false data injection attacks on power grid by sparse optimization," IEEE Transactions on Smart Grid, vol. 5, no. 2, pp. 612-621, 2014.

[10] C. Gu, P. Jirutitijaroen, and M. Motani, "Detecting false data injection attacks in AC state estimation," IEEE Transactions on Smart Grid, vol. 6, no. 5, pp. 2476-2483, 2015.

[11] D. He, S. Chan, and M. Guizani, "User privacy and data trustworthiness in mobile crowd sensing," IEEE Wireless Communications, vol. 22, no. 1, pp. 28-34, 2015.

[12] X. Lv and H. Li, "Secure group communication with both confidentiality and non-repudiation for mobile ad-hoc networks," IET Information Security, vol. 7, no. 2, pp. 61-66, 2013.

[13] B. Zhu, Z. Wan, M.S. Kankanhalli, et al., "Anonymous secure routing in mobile ad-hoc networks," in 2004 IEEE International Conference on Local Computer Networks (LCN), vol. 37, pp. 102-108, IEEE, 2004.

[14] Y. Liu, G. Liu, C. Cheng, et al., "A privacy-preserving health data aggregation scheme," KSII Transactions on Internet and Information Systems, vol. 10, no. 8, pp. 3852-3864, 2016.

[15] G. Liu, Y. Liu, C. Liu, et al., "Improved convertible multi-authenticated encryption scheme," Journal of Information and Computational Science, vol. 12, no. 8, pp. 3231-3240, 2015.

[16] Y. Zhan, Y. Xia, Y. Liu, et al., "Time-sensitive data collection with incentive-aware for mobile opportunistic crowdsensing," IEEE Transactions on Vehicular Technology, vol. 66, no. 9, pp. 7849-7861, 2017.

[17] W. Wang and B. Bhargava, "Visualization of wormholes in sensor networks," in 2004 ACM 3rd Workshop on Wireless Security (WiSe), pp. 51-60, ACM, 2004.

[18] S. Djahel, F. Naitabdesselam, and Z. Zhang, "Mitigating packet dropping problem in mobile ad hoc networks: Proposals and challenges," IEEE Communications Surveys and Tutorials, vol. 13, no. 4, pp. 658-672, 2011.

[19] M. Sasabe and T. Takine, "Continuous-time analysis of the simple averaging scheme for global clock synchronization in sparsely populated MANETs," IEEE Journal on Selected Areas in Communications, vol. 31, no. 4, pp. 782-793, 2013.

[20] D. Zhou and T.H. Lai, "An accurate and scalable clock synchronization protocol for IEEE 802.11-based multihop ad hoc networks," IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 12, pp. 1797-1808, 2007.

[21] M. Strasser, B. Danev, and S. Apkun, "Detection of reactive jamming in sensor networks," ACM Transactions on Sensor Networks, vol. 7, no. 2, pp. 16, 2010.

[22] W. Xu, K. Ma, W. Trappe, et al., "Jamming sensor networks: Attack and defense strategies," IEEE Network, vol. 20, no. 3, pp. 41-47, 2006.

[23] W. Xu, W. Trappe, Y. Zhang, et al., "The feasibility of launching and detecting jamming attacks in wireless networks," in 2005 ACM 6th International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), pp. 46-57, ACM, 2005.

[24] M. Spuhler, D. Giustiniano, V. Lenders, et al., "Detection of reactive jamming in DSSS-based wireless communications," IEEE Transactions on Wireless Communications, vol. 13, no. 3, pp. 1593-1603, 2014.

[25] S. Misra, R. Singh, and S.V.R. Mohan, "Information warfare-worthy jamming attack detection mechanism for wireless sensor networks using a fuzzy inference system," Sensors, vol. 10, no. 4, pp. 3444-3479, 2010.

[26] E. Sasikala and N. Rengarajan, "An intelligent technique to detect jamming attack in wireless sensor networks (WSNs)," International Journal of Fuzzy Systems, vol. 17, no. 1, pp. 76-83, 2015.

[27] A. Hamieh and J. Ben-Othman, "Detection of jamming attacks in wireless ad hoc networks using error distribution," in 2009 IEEE International Conference on Communications (ICC), pp. 1-6, IEEE, 2009.

[28] J. Soryal and T. Saadawi, "IEEE 802.11 DoS attack detection and mitigation utilizing cross layer design," Ad Hoc Networks, vol. 14, no. 3, pp. 71-83, 2014.

[29] J. Soryal, X. Liu, and T. Saadawi, "DoS detection in IEEE 802.11 with the presence of hidden nodes," Journal of Advanced Research, vol. 5, no. 4, pp. 415-422, 2014.

[30] P. Kyasanur and N.H. Vaidya, "Selfish MAC layer misbehavior in wireless networks," IEEE Transactions on Mobile Computing, vol. 4, no. 5, pp. 502-516, 2005.

[31] A.A. Cardenas, S. Radosavac, and J.S. Baras, "Detection and prevention of MAC layer misbehavior in ad hoc networks," in 2004 2nd ACM workshop on Security of Ad Hoc and Sensor Networks (SASN), pp. 17-22, ACM, 2004.

[32] S. Radosavac, A.A. Cardenas, J.S. Baras, et al., "Detecting IEEE 802.11 MAC layer misbehavior in ad hoc networks: Robust strategies against individual and colluding attackers," Journal of Computer Security, vol. 15, no. 1, pp. 103-128, 2007.

[33] S. Choi, D.Y. Kim, D.H. Lee, et al., "WAP: Wormhole attack prevention algorithm in mobile ad hoc networks," in 2008 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC), pp. 43-348, IEEE, 2008.

[34] E.S. Page, "Continous inspection schemes," Biometrika, vol. 41, no. 1/2, pp. 100-115, 1954.

[35] G. Lorden, "Procedures for reacting to a change in distribution," The Annals of Mathematical Statistics, vol. 42, no. 6, pp. 1897-1908, 1971.

[36] S. Zheng, T. Jiang, and J.S. Baras, "Performance comparison of two sequential change detection algorithms on detection of in-band wormholes" in 2009 IEEE 43rd Annual Conference on Information Sciences and Systems (CISS), pp. 270-275, IEEE, 2009.

[37] M.X. Cheng, Y. Ling, and W.B. Wu, "In-band wormhole detection in wireless ad hoc networks using change point detection method," in 2016 IEEE International Conference on Communications (ICC), pp. 1-6, IEEE, 2016.

[38] S. Xu and R.V. Boppana, "On mitigating in-band wormhole attacks in mobile ad hoc networks," in 2007 IEEE International Conference on Communications (ICC), pp. 1136-1141, IEEE, 2007.

[39] Y.C. Hu, A. Perrig, and D.B. Johnson, "Packet Leashes: A defense against wormhole attacks in wireless ad hoc networks," in 2003 IEEE International Conference on Computer Communications (INFOCOM), vol. 3, pp. 1976-1986, IEEE, 2003.

[40] Y.C. Hu, A. Perrig, and D.B. Johnson, "Wormhole attacks in wireless networks," IEEE Journal on Selected Areas in Communications, vol. 24, no. 2, pp. 370-380, 2006.

[41] M. Davison, "Multidimensional scaling," John Wiley and Sons, 1983.

[42] W. Torgeson, "Multidimensional scaling of similarity," Psychometrika, vol. 30, no. 4, pp. 379-393, 1965.

[43] A.M. Ladd, K.E Bekris, A. Rudys, et al., "Robotics-based location sensing using wireless ethernet," Wireless Networks, vol. 11, no. 1-2, pp. 189-204, 2005.

[44] R. Maheshwari, J. Gao, and S.R. Das, "Detecting wormhole attacks in wireless networks using connectivity information," in 2007 IEEE 26th International Conference on Computer Communications (INFOCOM), pp. 107-115, IEEE, 2007.

[45] X. Ban, R. Sarkar, and J. Gao, "Local connectivity tests to identify wormholes in wireless networks," in 2011 ACM 12th International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), pp. 13, ACM, 2011.

[46] S.J. Lee and M. Gerla, "Split multipath routing with maximally disjoint paths in ad hoc networks," in 2001 IEEE International Conference on Communications (ICC), pp. 3201-3205, IEEE, 2001.

[47] N. Song, L. Qian, and X. Li, "Wormhole attacks detection in wireless ad hoc networks: A statistical analysis approach," in 2005 IEEE 25th International Parallel and Distributed Processing Symposium (IPDPS), pp. 8, IEEE, 2005.

[48] M.Y. Su, "WARP: A wormhole-avoidance routing protocol by anomaly detection in mobile ad hoc networks," Computers and Security, vol. 29, no. 2, pp. 208-224, 2010.

[49] S. Hazra and S.K. Setua, "Rushing attack defending context aware trusted AODV in ad-hoc network," International Journal of Security, Privacy and Trust Management, vol. 1, no. 3, pp. 176, 2012.

[50] Y.C. Hu, A. Perrig, and D.B. Johnson, "Rushing attacks and defense in wireless ad hoc network routing protocols," in 2004 ACM 2nd Workshop on Wireless Security (WiSec), pp. 30-40, ACM, 2004.

[51] L. Tamilselvan and V. Sankaranarayanan, "Solution to prevent rushing attack in wireless mobile ad hoc networks," in 2006 IEEE International Symposium on Ad Hoc and Ubiquitous Computing (ISAUHC), pp. 42-47, IEEE, 2006.

[52] AL. Shahrani and A. Saad, "Rushing attack in mobile ad hoc networks," in 2011 IEEE 3rd International Conference on Intelligent Networking and Collaborative Systems (INCoS), pp. 752-758, IEEE, 2011.

[53] H. Kim, R. Oliveira, B. Bhargava, et al., "A novel robust routing scheme against rushing attacks in wireless ad hoc networks," Wireless Personal Communications, vol. 70, no. 4, pp. 1-13, 2013.

[54] M.Y. Su, "Prevention of selective black hole attacks on mobile ad hoc networks through intrusion detection systems," Computer Communications, vol. 34, no. 1, pp. 107-117, 2011.

[55] M. Imran, F.A. Khan, H. Abbas, et al., "Detection and prevention of black hole attacks in mobile ad hoc networks," in 2014 International Conference on Ad-Hoc and Wireless Networks (AdHocNets), pp. 111-122, Springer, 2014.

[56] S. Biswas, T. Nag, and S. Neogy, "Trust based energy efficient detection and avoidance of black hole attack to ensure secure routing in MANET," in 2014 IEEE International Conference on Applications and Innovations in Mobile Computing (AIMoC), pp. 157-164, IEEE, 2014.

[57] C.W. Yu, T.K. Wu, R.H. Cheng, et al., "A distributed and cooperative black hole node detection and elimination mechanism for ad hoc networks," in 2007 Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), pp. 538-549, Springer, 2007.

[58] D. Djenouri and N. Badache, "Struggling against selfishness and black hole attacks in MANETs," Wireless Communications and Mobile Computing, vol. 8, no. 6, pp. 689-704, 2008.

[59] E. Gerhards-Padilla, N. Aschenbruck, P. Martini, et al., "Detecting black hole attacks in tactical MANETs using topology graphs," in 2007 IEEE 32nd Conference on Local Computer Networks (LCN), pp. 1043-1052, IEEE, 2007.

[60] M. Jahnke, M. Bussmann, S. Henkel, et al., "Components for cooperative intrusion detection in dynamic coalition environments," Research Establishment for Applied Sciences Wachtberg-Werthhoven, 2004.

[61] J. Tolle, M. Jahnke, N.G. Felde, et al., "Impact of sanitized message flows in a cooperative intrusion warning system," in 2006 IEEE Conference on Military Communications (MILCOM), pp. 1-7, IEEE, 2006.

[62] M.R. Babu and G. Usha, "A novel honeypot based detection and isolation approach (NHBADI) to detect and isolate black hole attacks in MANET," Wireless Personal Communications, vol. 90, no. 2, pp. 831-845, 2016.

[63] J. Sen, M.G. Chandra, S.G. Harihara, et al., "A mechanism for detection of gray hole attack in mobile ad hoc networks," in 2007 6th IEEE International Conference on Information, Communications and Signal Processing (ICICS), pp. 1-5, IEEE, 2007.

[64] X. Gao and W. Chen, "A novel gray hole attack detection scheme for mobile ad-hoc networks," in 2007 IEEE International Conference on Network and Parallel Computing Workshops (NPC), pp. 209-214, IEEE, 2007.

[65] S. Zhong, J. Chen, and Y.R. Yang, "Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks," in 2003 IEEE International Conference on Computer Communications (INFOCOM), vol. 3, pp. 1987-1997, IEEE, 2003.

[66] S. Gurung and S. Chauhan, "A novel approach for mitigating gray hole attack in MANET," Wireless Networks, pp. 1-15, 2016.

[67] N. Marchang and R. Datta, "Light-weight trust-based routing protocol for mobile ad hoc networks," IET Information Security, vol. 6, no. 2, pp. 77-83, 2012.

[68] S. Marti, T.J. Giuli, K. Lai, et al., "Mitigating routing misbehavior in mobile ad hoc networks," in 2000 ACM 6th International Conference on Mobile Computing and Networking (MobiCom), pp. 255-265, ACM, 2000.

[69] K. Balakrishnan, J. Deng, and P.K. Varshney, "TWOACK: Preventing selfishness in mobile ad hoc networks," in 2005 IEEE Wireless Communications and Networking Conference (WCNC), pp. 2137-2142, IEEE, 2005.

[70] C. Pu and S. Lim, "A light-weight countermeasure to forwarding misbehavior in wireless sensor networks: Design, analysis, and evaluation," IEEE Systems Journal, pp. 1-9, 2016.

[71] B. Xiao, B. Yu, and C. Gao, "CHEMAS: Identify suspect nodes in selective forwarding attacks," Journal of Parallel and Distributed Computing, vol. 67, no. 11, pp. 1218-1230, 2007.

[72] X. Li, R. Lu, X. Liang, et al., "Side channel monitoring: Packet drop attack detection in wireless ad hoc networks," in 2011 IEEE International Conference on Communications (ICC), pp. 1-5, IEEE, 2011.

[73] S.B. Lee and Y.H. Choi, "A resilient packet-forwarding scheme against maliciously packet-dropping nodes in sensor networks," in 2006 ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN), pp. 59-70, ACM, 2006.

[74] O.F.G. Duque, A.M. Hadjiantonis, G. Pavlou, et al., "Adaptable misbehavior detection and isolation in wireless ad hoc networks using policies," in 2009 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 242-250, IEEE, 2009.

[75] O. Gonzalez, G. Ansa, M.P. Howarth, et al., "Detection and accusation of packet forwarding misbehavior in mobile ad hoc networks," Journal of Internet Engineering, vol. 2, no. 8, pp. 181-192, 2008.

[76] O.F. Gonzalez, M. Howarth, and G. Pavlou, "Detection of packet forwarding misbehavior in mobile ad-hoc networks," in 2008 International Conference on Wired/Wireless Internet Communications (WWIC), pp. 302-314, Springer, 2008.

[77] O.F. Gonzalez, M. Howarth, and G. Pavlou, "An algorithm to detect packet forwarding misbehavior in mobile ad-hoc networks," in 2007 IFIP/IEEE 10th International Symposium on Integrated Network Management (IM), pp. 813-816, IEEE, 2007.

[78] F. Anjum and R. Talpade, "LiPaD: Lightweight packet drop detection for ad hoc networks," in 2004 IEEE 60th Vehicular Technology Conference (VTC), pp. 1233-1237, IEEE, 2004.

[79] M. Sarkar and D.B. Roy, "Prevention of sleep deprivation attacks using clustering," in 2011 IEEE 3rd International Conference on Electronics Computer Technology (ICECT), pp. 391-394, IEEE, 2011.

[80] P. Yi, Z. Dai, Y. Zhong, et al., "Resisting flooding attacks in ad hoc networks," in 2005 IEEE International Conference on Information Technology: Coding and Computing (ITCC), pp. 657-662, IEEE, 2005.

[81] A. Chaudhary, V.N. Tiwari, and A. Kumar, "A cooperative intrusion detection system for sleep deprivation attack using neuro-fuzzy classifier in mobile ad hoc networks," Computational Intelligence in Data Mining, vol. 2, pp. 345-353, 2015.

[82] T. Martin, M. Hsiao, D. Ha, et al., "Denial-of-service attacks on battery-powered mobile computers," in 2004 IEEE 2nd Conference on Pervasive Computing and Communications (PerCom), pp. 309-318, IEEE, 2004.

[83] C. Piro, C. Shields, and B. Levine, "Detecting the sybil attack in mobile ad hoc networks," in 2006 IEEE International Conference on Security and Privacy in Communication Networks (SecureComm), pp. 1-11, IEEE, 2006.

[84] M. Jamshidi, E. Zangeneh, M. Esnaashari, et al., "A lightweight algorithm for detecting mobile Sybil nodes in mobile wireless sensor networks," Computers and Electrical Engineering, 2016.

[85] K.F. Ssu, W.T. Wang, and W.C. Chang, "Detecting sybil attacks in wireless sensor networks using neighboring information," Computer Networks, vol. 53, no. 18, pp. 3042-3056, 2009.

[86] S. Abbas, M. Merabti, D. Llewellyn-Jones, et al., "Lightweight sybil attack detection in MANETs," IEEE Systems Journal, vol. 7, no. 2, pp. 236-248, 2013.

[87] K. Geetha and N. Sreenath, "Detection of SYN flooding attack in mobile ad hoc networks with AODV protocol," Arabian Journal for Science and Engineering, vol. 41, no. 3, pp. 1161-1172, 2016.

[88] S. Wang, Q. Sun, H. Zou, et al., "Detecting SYN flooding attacks based on traffic prediction," Security and Communication Networks, vol. 5, no. 10, pp. 1131-1140, 2012.

[89] M. Bellaiche and J.C. Gregoire, "SYN flooding attack detection based on entropy computing," in 2009 IEEE Global Telecommunications Conference (GLOBECOM), pp. 1-6, IEEE, 2009.

[90] V.A. Siris and F. Papagalou, "Application of anomaly detection algorithms for detecting SYN flooding attacks," in 2004 IEEE Global Telecommunications Conference (GLOBECOM), pp. 2050-2054, IEEE, 2004.

[91] M. Korczynski, L. Janowski, and A. Duda, "An accurate sampling scheme for detecting SYN flooding attacks and portscans," in 2011 IEEE International Conference on Communications (ICC), pp. 1-5, IEEE, 2011.

[92] X. Long and B. Sikdar, "Wavelet based detection of session hijacking attacks in wireless networks," in 2008 IEEE Global Communications Conference (GLOBECOM), pp. 1-5, IEEE, 2008.

[93] X. Long and B. Sikdar, "A mechanism for detecting session hijacks in wireless networks," IEEE Transactions on Wireless Communications, vol. 9, no. 4, pp. 1380-1389, 2010.

[94] S.M. Glass, V. Muthukkumarasamy, and M. Portmann, "Detecting man-in-the-middle and wormhole attacks in wireless mesh networks," in 2009 IEEE International Conference on Advanced Information Networking and Applications (AINA), pp. 530-538, IEEE, 2009.

[95] B. Aziz and G. Hamilton, "Detecting man-in-the-middle attacks by precise timing," in 2009 IEEE 3rd International Conference on Emerging Security Information, Systems and Technologies (SECURWARE), pp. 81-86, IEEE, 2009.

[96] S. Brands and D. Chaum, "Distance-bounding protocols," in 1994 Workshop on the Theory and Application of Cryptographic Techniques (EUROCRYPT), pp. 344-359, Springer, 1994.

[97] I.H.E. De, G. Cochrane, J.M. Moreira-Lemus, et al., "Detecting and defeating advanced man-in-the-middle attacks against TLS," in 2014 IEEE 6th International Conference on Cyber Conflict (CyCon), pp. 209-221, IEEE, 2014.

[98] H.A. Kim and B. Karp, "Autograph: Toward automated, distributed worm signature detection," in 2004 Usenix Security Symposium (USENIX Security), pp. 271-286, USENIX, 2004.

[99] J. Newsome, B. Karp, and D. Song, "Polygraph: Automatically generating signatures for polymorphic worms," in 2005 IEEE Symposium on Security and Privacy (S&P), pp. 226-241, IEEE, 2005.

[100] S.A. Aljawarneh, R.A. Moftah, and A.M. Maatuk, "Investigations of automatic methods for detecting the polymorphic worms signatures," Future Generation Computer Systems, vol. 60, pp. 67-77, 2016.

[101] Y. Tang, B. Xiao, and X. Lu, "Using a bioinformatics approach to generate accurate exploit-based signatures for polymorphic worms," Computers and Security, vol. 28, no. 8, pp. 827-842, 2009.

[102] Y. Tang, B. Xiao, and X. Lu, "Signature tree generation for polymorphic worms," IEEE Transactions on Computers, vol. 60, no. 4, pp. 565-579, 2011.

[103] L. Wang, Z. Li, Y. Chen, et al., "Thwarting zero-day polymorphic worms with network-level length-based signature generation," IEEE/ACM Transactions on Networking, vol. 18, no. 1, pp. 53-66, 2010.

[104] B. Sun, K. Wu, and U.W. Pooch, "Routing anomaly detection in mobile ad hoc networks," in 2003 IEEE 12th International Conference on Computer Communications and Networks (ICCCN), pp. 25-31, IEEE, 2003.

[105] B. Sun, K. Wu, Y. Xiao, et al., "Integration of mobility and intrusion detection for wireless ad hoc networks," International Journal of Communication Systems, vol. 20, no. 6, pp. 695-721, 2007.

[106] N. Ye, X. Li, Q. Chen, et al., "Probabilistic techniques for intrusion detection based on computer audit data," IEEE Transactions on Systems, Man, and, Cybernetics, vol. 31, no. 4, pp. 266-274, 2001.

[107] A. Nadeem and M. Howarth, "Adaptive intrusion detection and prevention of denial of service attacks in MANETs," in 2009 ACM International Conference on Wireless Communications and Mobile Computing: Connecting the world wirelessly (IWCMC), pp. 926-930, ACM, 2009.

[108] H.O. Lancaster, "The chi-squared distribution," Journal of the Operational Research Society, vol. 27, no. 1, pp. 238, 1971.

[109] L.A. Doty, "Statistical process control," Industrial Press Inc.,U.S., 1996.

[110] Y. Zhang and W. Lee, "Intrusion detection in wireless ad-hoc networks," in 2000 IEEE 6th Annual International Conference on Mobile Computing and networking (MobiCom), pp. 275-283, ACM, 2000.

[111] Y. Zhang, W. Lee, and Y.A. Huang, "Intrusion detection techniques for mobile wireless networks," Wireless Networks, vol. 9, no. 5, pp. 545-556, 2003.

[112] D. Sterne, P. Balasubramanyam, D. Carman, et al., "A general cooperative intrusion detection architecture for MANETs," in 2005 IEEE 3rd International Workshop on Information Assurance (IWIA), pp. 57-70, IEEE, 2005.

[113] K. Ilgun, R.A. Kemmerer, and P.A. Porras, "State transition analysis: A rule based intrusion detection approach," IEEE Transactions on Software Engineering, vol. 21, no. 3, pp. 181-199, 1995.

[114] G. Vgina, S. Gawalani, K. Srinivasan, et al., "An intrusion detection tool for AODV based ad hoc wireless networks," in 2004 IEEE 20th Annual Computer Security Application Conference (ACSAC), pp. 16-27, IEEE, 2004.

[115] A.B. Smith, "An examination of intrusion detection architecture for wireless ad-hoc networks," in 2001 5th National Colloquium for Information System Security Education (CISSE), pp. 44-60, McGraw-Hill/Irwin, 2001.

[116] H. Tseng, T.Song, P. Balasubramanyam, et al., "A specification-based intrusion detection model for OLSR," in 2005 International Symposium on Recent Advances in Intrusion Detection (RAID), pp. 330-350, Springer, 2005.

[117] J.M. Orset, B. Alcalde, and A.R. Cavalli, "An EFSM-based intrusion detection system for ad hoc networks," in 2005 International Conference on Automated Technology for Verification and Analysis (ATVA), pp. 400-413, Springer, 2005.

[118] T. Clausen and P. Jacquet, "IETF RFC 3626: Optimized link state routing protocol (OLSR)," The Internet Society, http://www.ietf.org/rfc/rfc3626.txt, 2003.

[119] B. Alcalde, A. Cavalli, D. Chen, et al., "Network protocol system passive testing for fault management - a backward checking approach," in 2004 International Conference on Formal Techniques for Networked and Distributed Systems (FORTE), pp. 150-166, Springer, 2004.

[120] C.Y. Tseng, P. Balasubramanyam, C. Ko, et al., "A specification-based intrusion detection system for AODV," in 2003 ACM 1st Workshop on Security of Ad Hoc and Sensor Networks (SASN), pp. 125-134, ACM, 2003.

[121] H.C. Lin, M.K. Sun, H.W. Huang, et al., "A specification-based intrusion detection model for wireless ad hoc networks," in 2012 IEEE 3rd International Conference on Innovations in Bio-Inspired Computing and Applications (IBICA), pp. 252-257, IEEE, 2012.

[122] N. Stakhanova, S. Basu, Z. Wensheng, et al., "Specification synthesis for monitoring and analysis of MANET protocols," in 2007 IEEE International Conference on Advanced Information Networking and Applications Workshops (AINA), pp. 183-187, IEEE, 2007.

[123] J.F.C. Joseph, A. Das, B.C. Seet, et al., "CRADS: Integrated cross layer approach for detecting routing attacks in MANETs," in 2008 IEEE Wireless Communication and Networking Conference (WCNC), pp. 1525-1530, IEEE, 2008.

[124] A. Nadeem and M. Howarth, "A generalized intrusion detection and prevention mechanism for securing MANETs," in 2009 IEEE International Conference on Ultra Modern Telecommunications and Workshops (ICUMT), pp. 1-6, IEEE, 2009.

[125] A. Nadeem and M. Howarth, "Protection of MANETs from a range of attacks using an intrusion detection and prevention system," Telecommunication Systems, vol. 52, no. 4, pp. 2047-2058, 2013.

[126] K. Sanzgiri, B. Dahill, B.N. Levine, et al., "A secure routing protocol for ad hoc networks," in 2002 IEEE 10th International Conference on Network Protocols (ICNP), pp. 78-87, IEEE, 2002.

[127] P. Yi, Y. Zhong, and S. Zhang, "Distributed intrusion detection for mobile ad hoc networks," in 2005 IEEE/IPSJ International Symposium on Applications and the Internet Workshops (SAINT), IEEE, 2005.

[128] F. Alam, R. Mehmood, I. Katib, et al., "Data fusion and IoT for smart ubiquitous environments: A survey," IEEE Access, vol. 5, no. 99, pp. 9533-9554, 2017.