DSSAM: Digitally Signed Secure Acknowledgement Method for Mobile Ad-hoc Network

Ashutosh Srivastava
Indian Institute of Technology BHU Varanasi
Sachin Kumar Gupta (sachin.rs.eee@iitbhu.ac.in)
Shri Mata Vaishno Devi University https://orcid.org/0000-0001-8270-5853
Mohd Najim
University of Jeddah
Nitesh Sahu
Indian Institute of Technology Delhi
Geetika Aggarwal
Nottingham Trent University
Bireshwar Dass Mazumdar
Institute of Engineering and Rural Technology

Research

Keywords: Mobile Ad-hoc Network, Digital Signature, RSA, DSSAM, DSR, 2-ACK, Attacks, PDF, Routing Overhead

Posted Date: January 7th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-40371/v2

Version of Record: A version of this preprint was published on January 22nd, 2021. See the published version at https://doi.org/10.1186/s13638-021-01894-7.

DSSAM: Digitally Signed Secure Acknowledgement Method for Mobile Ad-hoc Network

Ashutosh Srivastava[1], Sachin Kr Gupta[2*], Mohd Najim[3], Nitesh Sahu[4], Geetika Aggarwal[5], Bireshwar Dass Mazumdar[6]

[1]Department of Electrical Engineering,
Indian Institute of Technology (BHU), Varanasi-221005, (Uttar Pradesh), India

[2]School of Electronics and Communication Engineering,
Shri Mata Vaishno Devi University, Kakryal-182320, Katra (Jammu & Kashmir), India

[3]University of Jeddah, College of Engineering,
Department of Electrical and Electronics Engineering, Jeddah-21589, Saudi Arabia

[4]Centre for Applied Research in Electronics,
Indian Institute of Technology, Delhi-110016, India

[5]School of Science and Technology,
Nottingham Trent University, Nottingham, NG11 8NS, UK

[6]Department of Computer Science and Engineering,
Institute of Engineering and Rural Technology, Prayagraj-211002, (Uttar Pradesh), India

Email Id: ashutosh.rs.eee@iitbhu.ac.in, sachin.gupta@smvdu.ac.in, mngalib@uj.edu.sa, nitesh.sahu@care.iitd.ac.in, geetika.aggarwal@ntu.ac.uk, bireshwardm@gmail.com

Abstract

Mobile Ad-hoc Networks (MANETs) is an infrastructure-less, self-motivated, arbitrary, self-configuring, rapidly changing, multi-hop network that is self-possessing wireless bandwidth-conscious links without centrally managed router support. In such a network, wireless media is easy to snoop. It is firm to the surety to access any node, easier to insertion of bad elements or attackers for malicious activities in the network. Therefore, security issues become one of the significant considerations for such kind of networks. The deployment of an effective intrusion detection system is important in order to provide protection against various attacks. In this paper, a Digitally Signed Secure Acknowledgement Method (DSSAM) with the use of the RSA digital signature has been proposed and simulated. Three different parameters are considered viz. secure acknowledgment, node authentication and packet authentication for study. This article observes the DSSAM performance and compares it with two existing standard methods, namely, Watchdog and 2-ACK under standard Dynamic Source Routing (DSR) routing environment. In the end, it is noticed that the rate of detection of malicious behaviour is better in the case of the proposed method. However, associated overheads are high. A trade-off between performance and overhead has been considered.

Keywords: Mobile Ad-hoc Network; Digital Signature; RSA; DSSAM; DSR; 2-ACK; Attacks; PDF; Routing Overhead.

1. Introduction

The MANETs is a decentralized kind of network, where nodes of the network relay packet to each other on the concept of the store and forward, i.e. nodes may also act as routers finding and maintaining routes to one another. Here, nodes can participate freely and leave without centralized control. Generally, due to the varying velocity of mobile nodes, the network topology may variate arbitrarily and rapidly in an irregular way. Therefore, the phenomenon of frequent link breakage is quite common. The moments of nodes are independent of one another, unlike others which use committed nodes to endorse functions such as network management, packet forwarding, and routing [1]. These functions are distributed to all available nodes by the ad-hoc networks since the ambiance causes the nodes to be easily captured and compromised. Hence, it is essential to provide security measures [2, 3]. Therefore, security in MANET is a crucial consideration. In addition, the routing of operations could also be easily compromised if safety measures are not integrated into the network functions.

In general, in MANETs, routing protocols are designed with assumptions that every participating node will fully cooperate with each other. This network does not have any type of centrally administrative services. All networks that function such as network control, routing, forwarding packets, including switching, etc. are communicated between terminals (nodes) either in cooperation or independently. Therefore, coordination between nodes is rather solicited. However, due to its transparent characteristics and restricted on-hand battery power of nodes, malicious activities can also be done in this network. Moreover, the MANETs structure may differ based on their various applications from static, small to dynamic, highly mobile in nature (vehicular, FANET, etc), and large-scale network which is highly energy constrained [4, 5].

In the MANET environment, the array of mobile wireless nodes is interconnected either for generic aims such as time-critical applications like tactical, law enforcement and emergency operations or for distinct goals like only shares their resources for ensuring global connectivity [6]. However, few resources, for example battery power are consumed rapidly as participating nodes have to perform network functioning tasks. When a node's power is taken as one of the most significant in such an environment, so there may chances that nodes may deny sharing their own resources in order to save battery power or to get benefit from other nodes [8]. These participating nodes are termed as misbehaving or selfish nodes and their activities are called misbehaviour or selfishness [10]. This kind of network is a cooperative network. So, in order to provide good cooperation among participant nodes, an already significant amount of control overheads packets is needed. Therefore, security measures are generally not implemented in the protocols to keep the overhead low, i.e. nodes are not checked for maliciousness. Due to this reason, MANETs are easy

targets for attackers. The attackers perform the malicious activity in one and most common way by injecting non-cooperative nodes into the network. Therefore, the development and implementation of the intrusion detection system become one of the prime duties in this network.

Already, various techniques [7, 9, 11-20] have further presented in the literature study in order to identify and reduce the effect of such misbehaviour or selfish nodes in a MANET, and VANET (vehicular ad-hoc network) environment. I.e. inspections of past works cover intrusion detection and prevention techniques. Many of these techniques have been evaluated based on performance metrics and routing schemes of MANETs. Among various techniques, Watchdog, Pathrater, and 2-ACK [11, 23] are highlighted one, which can significantly identify and reduce the impact of network maliciousness, respectively. Watchdog provides the mechanism to recognize bad elements in the network by overhearing the wireless transmission media and is the passive type of overhearing method, while, the Pathrater technique does not allow malicious nodes to participate in the process of route determination. 2-ACK security scheme reduces the bad effect of such immoral elements. From a previously reported works, one can observe that still various issues like obscure and receiver collision, false behaviour, limited transmission range, etc. still need to be addressed and can be considered as a weakness of most highlighted security techniques.

Our proposed system uses the cryptographic mechanism to make the network secure and try to overcome the above-mentioned weakness. Three important security aspect of MANET has been considered viz. secure acknowledgment, node authentication, and packet authentication. Our presented DSSAM performs better, in the sense of identification of malicious nodes and its activities, but with the cost of the significant amount of overheads.

DSSAM is well suited in high level use of various Internet of things (IoT) application scenarios where the proposal will be applicable as security solution in terminal to terminal communications at hybrid ad-hoc network solutions. Actually, IoT is the next eon of communication in which physical objects can be empowered to create, receive and exchange data in a seamless manner with heterogeneous network environment also. The various IoT applications focus on automating different tasks and are trying to empower the inanimate physical objects to act without any human intervention. The existing and upcoming IoT applications are highly promising to increase the level of comfort, efficiency, and automation for the users and for such environment. To be capable to gizmo such an ecosphere in a constantly emergent approach requires better and high security, authentication, privacy and recovery from attacks. In this respect, it is imperious to make the required modifications in the design of IoT applications for achieving secure IoT atmospheres. In this paper, a detailed discussion and improvement over watchdog to 2-ACK and then 2-ACK to DSSAM method is explained with considering few performance metrics. The proposed DSSAM approach will

help to achieve a high degree of trust and increase the level of security in the potential useful IoT applications with hybrid environment such as:

a. Smart transportation system.

b. Smart agriculture and animal farming.

c. Smart emergencies environment.

d. Smart communication at defence scenario.

e. Smart commercial, residential and Industrial area, and many more.

1.1 Motivations and Principal Contributions

Since the last few decades, the outlook of wireless networking is drastically changing due to fast growth in wireless technologies and requirements of new wireless services and various applications as well. The wireless industries have experienced unexcelled growth, from satellite broadcasts into countless households to Wireless Personal Area Networks (WPAN) [13], VANET [15], WSN [16], etc. Consequently, the cost of wireless access falls; hence it can replace wired access in many aspects. One of the greatest advantages of wireless is to provide connectivity among users while roaming. However, the distance between users is limited due to the short distance of transmitter or their vicinity to Wireless Access Point (WAP) [13]. Later, in the 70's onward era, the development of MANET has overcome this problem by involving intermediate nodes to forward data packets to the outside range of nodes [1-2].

One of the most vibrant and rapidly growing fields nowadays is the MANET. It is also called as the wireless mobile multi-hop or mobile packet radio network. In this realm, significant research is going on since last nearly fifty years in order to its betterment. Due to infrastructure-less, self-configuring, self-motivated properties of MANET, it has got possible future applications in different fields such as tactical environments, emergency operations, home and enterprise, commercial, civilian environments, traffic environment [19], location-aware services, and extension of coverage, etc [8, 14]. This network is vulnerable due to its important features such as distributed service, open medium, autonomous terminal, dynamic topology, lightweight terminals, asymmetrical communication, fluctuating link capacity and constrained capability, etc [27]. These above fundamental characteristics introduce several challenges for researchers in the MANET environment, where security issue is one of the significant issues. MANET can maximize its Quality of Service (QoS) parameters such as throughput, Packet Delivery Fraction (PDF), etc. by using all the intermediate nodes accessible to route and then forwarding packets. However, the node can consequently behave badly by refusing to supply providers or shedding down the packets in the

community due to the fact of its selfishness, malicious exercise, etc. [28-29]. Identifying and preventing misbehaving nodes from them can be one of the biggest challenges for a network like that. The principal contributions of the current research article are as follows:

a. State-of-the-art of various user authentication schemes as well as intrusion detection strategies, have been analysed for the MANET and WSN environment.

b. The MANET application layer has attracted vast research as well as the scientific community during the last few decades. As a result, many user authentication techniques for MANET and WSN have been proposed and published in the literature. Among them, a few most closely relevant to our proposed method are explored.

c. Article also discusses the possible security attacks on different security goals along with its target and prevention schemes.

d. Due to open and decentralized characteristics of MANET, misbehaving, or the suspicious nodes may be involved in the process of route discovery. Further, they may refuse to provide the information/services in the network, i.e. deny forwarding the data packets. Therefore, this article tries to identify the existing intrusion detection systems that can identify and prevent disruptive network operations.

e. Existing intrusion detection techniques such as Watchdog and 2-ACK are explored in terms of their strength and weakness.

f. To provide secure authentication and an acknowledgment mechanism in MANET, we proposed DSSAM that is based on RSA digital signature. This scheme overcomes the weakness of existing intrusion detection techniques such as receiver collision, false identity problem, etc.

g. Finally, the proposed authentication approach has been compared with the current techniques.

This research article is structured as follows: immediate subsequent section presents background with a literature survey on co-related work in this area followed by a discussion of intrusion detection techniques in the next section. Moreover, after that digital signature with its needs, including signature creation and verification steps have been discussed in the next section followed by problem definition and the proposed method. Further, performance evaluations of DSSAM, Watchdog, and 2-ack have been made through a simulation study followed by results and discussion. At last, it comes to its conclusion and possible future scope.

2. Literature Survey

The conveyed work in the state-of-the-art of secure acknowledgment in MANET, WSN and related domain by several scientists and researchers has been presented in this section.

The work in [23] explained routing misbehaviour in MANETs and suggested a 2-ACK technique for identifying and minimizing the impact of selfish nodes in the routing. 2-ACK is based on a simple 2-hop acknowledgment packet that is returned by the next-hop link recipient. The 2-ACK mechanism operates as an alternative routing scheme strategy for detecting routing misconduct and reducing its adverse effects. The 2-ACK mechanism solves several problems, including limited transmission powers, ambiguous collisions and receiver collisions. The 2-ACK scheme can be used efficiently in DSR in MANET. Trust Aware Routing Protocol (TARP) as an advanced security routing mechanism based on the level of trust was presented and evaluated [24]. TARP is a technique that allows for the search of safe routes in MANET. The authors measured the trust parameter based on a defined set of parameters and used it in TARP. The study shows that TARP will improve an ad-hoc network's defense and rising routing congestion while preserving a reasonable route discovery period and an appropriate pause. The routing traffic relates specifically to the collection of nodes that meet the sender's requirements. Two techniques of Watchdog and Pathrater are explained in [11] that helps to increase ad-hoc network throughput. Both methods are extensions of DSR algorithms to reduce the impact of ad-hoc network routing misconduct. Watchdog identifies nodes that are misbehaved, and the Pathrater strategy helps to redirect protocols to prevent packet movement of those nodes. The yield of these two strategies improves the efficiency of a relatively mobile network by 17 percent, thus growing the ratio of overhead transmission to data transmission from 9 percent to 17 percent of the regular routing protocol.

The black hole attacks are a serious problem widespread in mobile ad-hoc networks [25]. Work focuses on the vulnerabilities of MANET and it looks at the black hole attacks. They portrayed the creation of an enhanced algorithm called Radical Watchdog and Pathrater for recognizing and removing Black Hole Attacks. In the article [26], the authors introduced a scheme called cluster-based trust to alleviate the internal attacks. In this research, the network is divided into cluster groups. Every cluster is certified as having the cluster head. The node decides the trust value and delivers it to the head of the cluster for their one-hop neighbours. In addition, the cluster head gives its participant nodes the certificate of confidence. This mechanism gives a good fraction of packet delivery and resilience to internal attacks. A novel technique is proposed to secure MANETs by addressing network configuration and security issues during the response and recovery phase [27]. This work analysed the threats to security and presented the security goals to be achieved and set up a stable key management system in an ad-hoc communication environment. A MANET-based algorithm for effective security and trust management is provided in [28]. In the sense that the produced nonce is not easily detectable, the time-based nonce is produced at specific time intervals that give the suggested approach reliability. It has been compared with the already existing trust-

based approach and finds better detection performance of the security threat in MANET. Several techniques are discussed in [29], for example, reverse engineering, repacking, and hex editing to circumvent the host anti-virus signatures. Comprehensive comparison studies were conducted of various methods where malware could get the hosts from outside of the networks. A new honey-net based intrusion detection technique is also discussed. In MANETs, a complete survey of Intrusion Detection Systems (IDSs) is well presented in [30, 31]. They categorize the architectures for intrusion detection framework in the MANET, and each one is ideal for evaluating and comparing various network infrastructures on node cooperation. Similarly in another research [21], authors proposed pseudonym generation based genetic algorithm to solve the location privacy problem in vehicular ad-hoc network, and thus guarantees un-traceability by an adversary. Further, authors of [22], study the physical layer security issues in vehicular environment. They shows that how the secrecy capacity and secrecy outage probability of a vehicular network can improved with respect to the source power, eavesdropper distance.

Due to vast applications of WSNs, it is ensuring that the only permitted availability of information is accessible via sensor nodes is often an open challenge. In this review work [32, 33], twenty-two features have been presented in which a secure user authentication mechanism should be in place, and then seven possible schemes were tested against the features specified. The analysis has been started from Wong's work [34] in 2006 and has been concluded at Vaidya *et al.*'s technique implemented in 2012 [35]. In each scheme the user impersonation and gateway nodes (GWN) bypass attacks and are likely. There is almost no scheme like that provides consumer confidentiality and repairability in case of failure or theft of smart cards. A scheme that only withstands an impersonation attack by a sensor node and a parallel session attack [36]. The replication attack and the fake verifier attack can only be taken on scheme suggested by Wong *et al.*'s and Tseng *et al.*'s in [34, 37]. Yoo *et al.*'s scheme offer mutual authentication between SN and GWN, and Khan-Alghatbar's scheme achieves success in mutual authentication between users and GWN and even SN and GWN [36, 38]. Just one scheme avoids DoS attack and offers hidden parameter protection to the gateway node. In short, no scheme is completely protected to all available features and all the strategies meet no authentication feature. The network communication security is one of the most important challenges in WSN [39]. HWSNs has optimized network capacity and introduced High-resource Network Sensor Nodes. An efficient adaptive authentication and key management schemes are being proposed for HWSNs in this article. The proposed protocol provides the authentication and key management for HWSNs along with optimization of security level, memory consumption, computational complexity, and overhead coordination which in effect enhances energy efficiency. The key distribution algorithm described here for producing dynamic keys focuses on pre-existing

information. Therefore, the exchange of keys does not involve a secure channel, and the process of sharing. Therefore, it increases security and energy efficiency.

We carry out an extensive literature review and make an analysis of the existing techniques for the identification and removal of different forms of attacks within the ad-hoc network. Our work culminates with the design of a digitally signed secure acknowledgment algorithm for enhanced security in the ad-hoc network. It aims to tackle Watchdog's restricted communication power and collision problems with receivers with better securing the system by securing acknowledgment, node authentication and packet authentication with digital signature technique.

3. Preliminary Studies

The presence of attackers in the network cannot be taken too lightly. Therefore, the basic functionality of different attacks that may impact the various securing schemes of MANET needs to be understood. In this section, few essential parameters such as security goals, attack models and usability attributes have been discussed. Moreover, this section also describes the various intrusion detection schemes like Watchdog, 2-Ack. These preliminary studies are indeed needed for a better understanding of our proposed security mechanism: DSSAM.

3.1 Security Goals, Security Attack Models and Usability Attributes:

This sub-section presents various attacks that are supposed to be resisted by MANETs. Also list outs various useful features that should offers by the proposed authentication method in order to provide an amicable and a reliable security mechanism. The different security goals such as confidentiality, integrity, availability, end to end authentication, etc. may be threatened by various security attacks [33, 40]. The comparative study of various security attacks in terms of their target and its prevention is illustrated in Table 1 [41].

3.1.1 Security Goals (SG): The different kinds of security goals are as given:

a. *SG1. Confidentiality:* All communicating individuals (i.e. approved parties) can understand the content of a message.

b. *SG2. Integrity:* Guarantee that the message received at another individual is the same as the message originally sent by the sender when the message is inserted into the network (i.e., the message will not be modified in any way).

c. *SG3. Availability:* Message shall be made accessible only to authorized entities.

d. *SG4. Authentication:* Guarantee that anyone sending or accessing the sensitive message has to be approved.

3.1.2 Security Attack (SA) Models: Figure 1 also shows different security threats as follows:

a. *SA1. Snooping:* This is a passive type of attack relating to unauthorized access or interception of communications content. SA1 may be prevented by using encipherment methods to make the content of communications non-intelligible.

b. *SA2.* Traffic Analysis: Such groups of attackers basically consider one communication pattern within the MANET environment.

- Network traffic monitoring: e.g. log files, Web pages, etc.
- Seek to obtain valuable statistical analytical information: e.g. who interacts with whom, where, for how long, where? And who cares about what content, etc.?

c. *SA3. Modification:* This is something of a deliberate kind of attack. Attackers attempt to change the information in order to make their own benefit after accessing the document. In this scenario, attackers also often seek to delete or interrupt the post, to harm or benefit the machine.

d. SA4. Masquerading: Masquerading or spoofing form of attack may be deployed on the ad-hoc mobile network while someone else is being impersonated by the attacker. Firstly, an intruder intercepts one or more legitimate authentication queries. Later, modify this request to allow it to pass MANET's authentication test and get authorization to access services inside the network.

e. SA5. Replaying: Anyway, in this SA model, the intruder receives a copy of a message received by the legitimate user to either access the MANET or trick the lawful user by claiming himself to be a genuine service provider. If an intruder fails, then it could be considered the assault as a replay defense threat.

f. *SA6. Repudiation:* It's something of a particular kind of attack from the one that has been mentioned before. SA6 is conducted by either source or destination on one of the two permitted communication parties within the MANET. The message sender denies later that he sent the message in this case, or the receiver can later deny that he received the message.

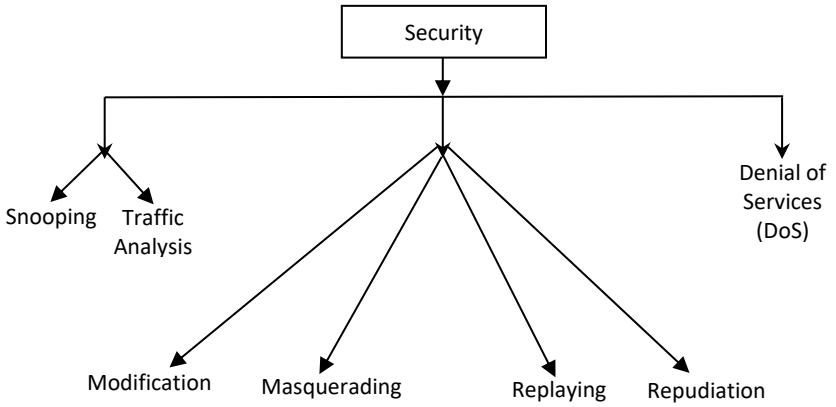


Fig. 1. Various Possible Security Attacks

g. *SA7.* DoS: It is an aggressive kind of attack and generally very normal. It can slow it down or completely disrupt a system/network service [30]. In this scenario, attackers may initiate several ways to reach the target. We can inject too many fake requests into the network that the server crashes due to the heavy traffic load. If the intruder succeeds in launching this attempt, then the node of MANET is irresponsive, and no one can link to it.

Table 1. Comparative Study of Various Security Attacks: Target and Prevention

Security Attacks (SA)	Active/ Passive	Threaten	Target of SA	Prevention
Snooping	Passive	Attacks Against Confidentiality/Privacy	Just to obtain the content of message	Data encipherment schemes
Traffic Analysis	Passive			
Modification	Active	Attacks Against Integrity	Modification of data exchanged between MANET users	Hash function to assure data integrity; SHA-1 or MD5
Masquerading	Active			
Replaying	Active			
Repudiation	Active	Attacks Against Integrity & Authentication	Modification of data & Disturbance of Source and Destination Authentication	Hash functions & Digital Signature Techniques
DoS	Active	Attacks Against Availability	Unavailability of services	Check identity & password of each users in MANET

3.1.3 Usability Attributes (UA): The proposed authentication MANET scheme also supports various usability attributes along with resistance capacity against different attacks. The several usability attributes are listed out in Table 2 with its descriptions.

Table 2. Important Usability Attributes of Authentication Mechanism with its Description

UA	Attributes' Name	Description
UA1	Authentication between MANET's user and intermediate node.	Since end-users may request for services or information, and MANET intermediate nodes provides services/information. Therefore, mutual authentication between end-users and intermediate node becomes essential.
UA2	Authentication between MANET's end users	Authentication should also be done between two end-users, i.e. the sender and receiver in order to avoid repudiation.
UA3	User's amicable registration phase	Registration phase from registration authority should be amicably and not put any kind of burden on users like remembering to the random number, etc. until not received the smart card.
UA4	Password change facility: user friendly and secure	Whenever the user needs to change the password, it should be friendly without interrupting registration authority. Keeping the same password for a long time is not suggested due to vulnerability towards password guessing. Moreover, the password changing mechanism must be secure in order to avoid false updating from the adversary.
UA5	Creation of session key	In order to provide good mutual authentication between two authorized communication parties, to keep confidentiality of messages and to provide a secure communication atmosphere after the end of the session generation of session key becomes an important concern.

3.2 Intrusion Detection Techniques in MANETs

Each node in MANETs presumes that other nodes work together to transmit and receive data. This paves the attackers the opportunity to respond and carry out the malicious operation with few compromised nodes on the network. To address this problem three important functions viz. prevention, detection, and recovery have been considered [31]. These functions provide three-layered security to MANETs. This section discusses the intrusion detection system usually the second security layer [32]. Two classical detection approaches, namely: 2-ACK and Watchdog.

3.2.1 Watchdog Method

The Watchdog methodology acts as a DSR extension. The feature named Watchdog that detects mischievous nodes; it has also built a component called Pathrater that calculates a path for these nodes to flee. Each node must execute certain modules on the network. Often Watchdog listens promiscuously for transmission of the next node. This also checks that the node is forwarding the received packet correctly. The Watchdog enables the feature of detection if the node has altered with the payload. The major question for this method is how it will perform, so the solution is to fit the listened packet to the freshly sent packet buffer. The Pathrater module processes data that the Watchdog receives to score the efficiency of any other node in the network knows and calculates a route metric derived by comparing the node scores in the route. The packets should then be routed through direction with the highest metric. This program can never be turned against the network because such conduct will be detected easily. Node X (mischievous) may falsely complain that node B does not forward packets in a route A-X-B-C-D. Nonetheless, acknowledgment of a message from A to D is moving accurately from D to A (Node X cannot leave packets or their acknowledgment, because both A & B will consider this malfeasance), and then A is conscious that B is not misconducting because it's part of the route.

Considering the name of the path as A -B -C. The drawback of this framework is that in some subsequent situations the Watchdog operating in node A may fail to identify a node that is misbehaving.

- There may be a packet clash in A when A is listening to B. In this scenario, A can not say if the co llision was triggered by B transmitting the packet (well-behaving) or by transmitting another node when B has not transmitted the packet (misbehaving);
- A listen to the B forwarding to C, it seems that B correctly transmits the packet. Node A, however, cannot determine whether it has been received by C or crash in C and B did not re-send (misbehaving) the packet.

- Node B can change its transmission capacity (misbehaving) to allow A to identify that B is transmitting a packet to C but that C is not receiving it.
- Nodes B and C (both of which are misbehaving) will cooperate with the launch of an attack. Node transfers a packet to C appropriately, but it does not say C drops the packet.
- Node B can lose packets at such a lower rate than A's Watchdog's minimum threshold for misbehaviour.

The above-described method can be better understood with the block diagram of Figure 2. It detects the misbehaving nodes [11]. Suppose a path runs from node S to D through A, B, and C. Still, A is not capable of transmitting to C but it can respond to B. So, A can tell if B broadcast the packet. If encryption is not conducted on each connection (which itself is an expensive and complex affair), then A can also say whether B has tampered with either payload or header.

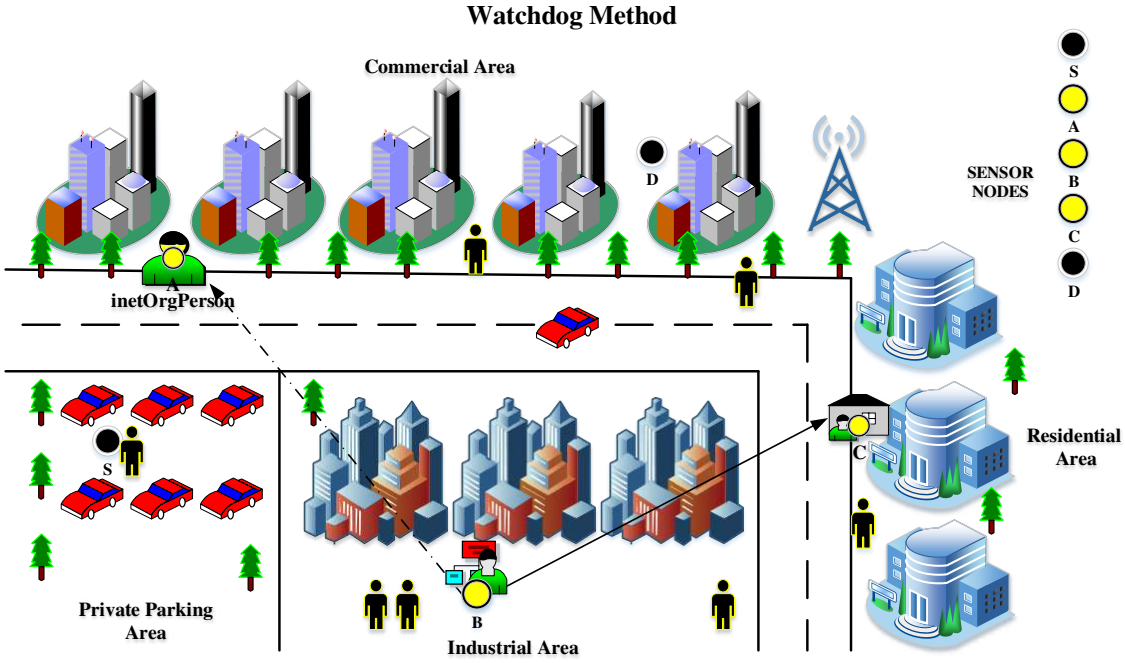


Fig. 2. Watchdog Method

The DSR routing protocol can identify misconduct at the forwarding point. The weakness of Watchdog lies in the fact that it may not be capable of detecting a node mistreating in the context of following collisions:

- Ambiguous collisions,
- Collisions with receiver,
- Limited transmission power,
- False misbehavior,
- Collision and partial dropping.

3.2.2 2-ACK method

It is a network layer strategy for detecting links that are misbehaving and mitigating their impact. This technique can be implemented as an extension to establish routing protocols such as DSR in MANETs already. A 2-ACK packet is assigned a fixed two hops path in the opposite way of the network traffic route. To overcome the weakness of Watchdog, Liu *et al* [23] proposed a 2-ACK method. It aims to overcome Watchdog's limited transmitting power and collision problems with receivers. It responds as acknowledges on each data packet transmitted over two hops distance and

all three consecutive nodes alongside the path from source to destination. In this way, it detects misbehaving links. Suppose three consecutive nodes (triplet) alongside a path are N1, N2, and N3. Node N1 will deliver packet 1 to N2, and N2 will deliver the same to N3.

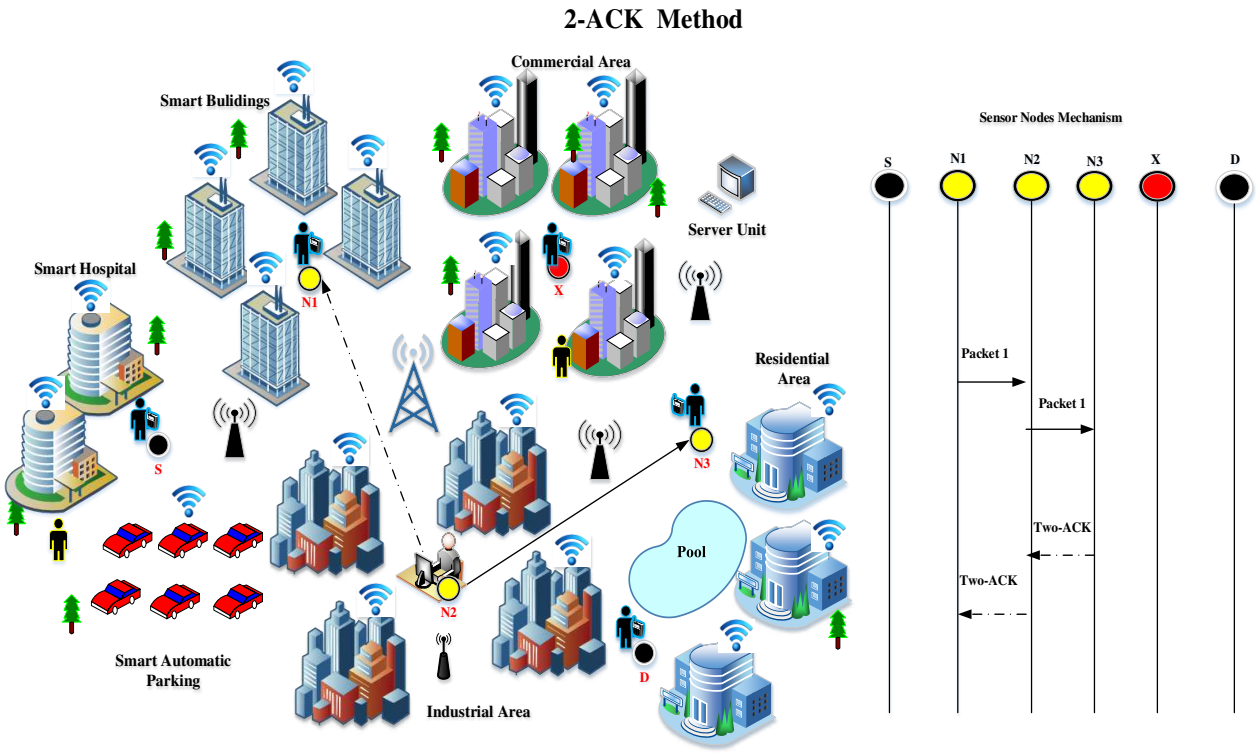


Fig. 3. 2- ACK Method

Upon receiving the packet, N3 generates a 2-ACK packet containing the reverse path between N3 and N1 and return to N1. This message, when received by N1, shows packet successfully communicated from N1 to N3 or else, if this 2-ACK packet is not delivered within a predefined time, all N2 and N3 nodes will be identified as malicious. The same procedure applies in the remaining route to each of the three successive nodes. A considerable amount of unfavourable overhead network was added to the acknowledgment process in order to process each packet transmission [42, 43].

The above method we can better understand with a block diagram and a more explicit working approach. Figure 3 exhibits the working model of the 2-ACK method. In the route discovery process of the MANETs DSR system, the path from a source node (S) to a destination node (D) finds out. When N1 delivers a data packet to N2, and N2 transfers it to N3, it is uncertain if N3 receives the data packet successfully or not. There is such confusion, even when no nodes are misbehaving. The problem gets even more serious in open MANETs with potential nodes that misbehaved. The 2-ACK scheme requires a clear acknowledgment from N3 to notify N1 of its positive reception of a data packet. If node N3 receives the data packet efficiently, it passes a 2-ACK packet to N1 over two hops (i.e. the opposite routing route direction, as shown) with the discovery of the associated data packet. The triplet [N1 →N2 →N3] comes from the direction of initial data traffic. N1 uses such a triplet to track the N2 → N3 connection. For display simplicity, we mark N1 as the 2-ACK packet recipient or the observer node and N3 as the 2-ACK packet sender in the triplet [N1 →N2 →N3]. For any group of triplets along the path such a 2-ACK connection happens. Consequently, only the first router of the source does not act as a 2-ACK packet sender just before arrival and destination the last router will not be functioning as 2-ACK receivers. The 2-ACK packet sender keeps a record of data packet IDs that were submitted but were not recognized for misbehaviour. For e.g., After N1 sends a data packet on a particular direction, say, [N1 → N2 → N3] shown in Figure 3, it attaches the data ID to LIST (see Figure 4, showing the data structure retained by the observing node), i.e. to its list corresponding to N2 → N3). At the same moment, a list of data packets transmitted, Cpkts, is incremented.

N2 Next Hop Receiver	N3 Second-Hop Receiver	Cpkts Packets Transmitted	Cmis 2-ACK packets Missed	LIST List of data Packet IDs

Fig. 4. Data Structure Maintain by Observing Node

Pseudo code for 2-ACK Method

We use the triplet *N*1 → *N*2 → *N*3 in Figure 2 as an example to illustrate 2-ACK's pseudo code. Note that such codes are run on each of the sender/receiver of the 2-ACK packets.

X.1 2-ACK Packet Sender Side (Node *N*3)

2: *Cpkts* ← 0, *Cack* ← 0, *i* ← *n* # *Initialization at node N*3
3: **while** *true* **do**
4: **if** (*data packet received*) **then**
5: *Cpkts* ++ # *Increase the counter of received packets*
6: **if** (*Cack* = *Cpkts* < *Rack*) **then** # *The data packet needs to be acknowledged*
7: *prepare MAC*
8: *prepare* 2-ACK *with ID*
9: *send* 2-ACK
10: *Cack* + +, *i* – – # *Increase the counter of acknowledged packets*
11: **end**
12: **end**
13: **end**

X.2 Receiver (Observer) Side (Node *N*1)

Parallel process 1 (receiving *hn*)

```
1: while true do
2: if receive hn from the 2-ACK packet sender then
3: record hn, i ← n
4: end
5: end
Parallel process 2 (receiving 2-ACK packets)
6: while true do
7: randomly select Tstart > current time  # Start the observation
8: while current time < Tstart do
9: # null
10: end
11: LIST ← φ, Cpkts ← 0, Cmis ← 0 # Initialization at node N1
12: while current time < Tstart + Tobs do # Observation period is not expired
13: if (data packet forwarded) then
14: LIST ← LIST U data ID # Add a data ID to LIST
15: Cpkts ++ # Increase the counter offorwarded packets
16: setup timer (τ) for data ID # Record the time
17: end
18: if (2-ACK packet received) then
19: search data ID carried by 2-ACK from LIST
20: if (found) then == A 2-ACK packet for a data ID received
21: check validity of hi
22: LIST ← LIST - data ID # Remove data ID from LIST
23: clear timer for ID
24: end
25: end
26: if (timeout event happens) then # 2-ACK packet for a data ID is not received
27: LIST ← LIST - data ID == Remove data ID from LIST
28: Cmis ++ # Increase misbehaviour counter
29: end
30: end
31: if (Cmis = Cpkts > Rmis) then # The observation period expires
32: send link misbehaviour report
33: end
34: end
```

Fig. 5. 2-ACK Executions Process

Each ID will remain on the list for τ seconds at N1, the reception timeout for 2-ACK. Before the expiration of the time if a 2-ACK packet matching to this ID, the ID will be deleted from the list. Alternatively, the ID would be deleted at the end of its timeout period, incrementing a counter called Cmis. Once N3 encounters a data packet, it determines if it will send a 2-ACK packet to N1. 2-ACK packets must accept only a fraction of the data packets to reduce the extra overhead routing caused by the 2-ACK method. Such a percentage is called the Ratio (Rack) identification factor. By adjusting the Rack, we can efficiently balance the overhead for 2-ACK packet transfers. Node N1 watches the behaviour of node N2 and N3 for a time called Tobs. At the end of the observation period, N1 calculates the sum of missing 2-ACK packets as Cmis / Cpkts and compares them with a Rmis threshold. When the ratio is greater than Rmis, it is deemed to be misbehavioural and N1 sends out a RERR packet (or misbehavioural notification). Since only a fraction of the obtained data packets is identified, Rmis will satisfy Rmis > (1–Rack) with the goal of removing false alarms triggered by such a partial acknowledgment technique. The node obtains or overhears such a RERR marks the N2 as misbehaving connection N3 and adds such misbehaving links to the blacklist it

maintains. If a node later begins its own data flow, it stops using these connections as part of its route as misbehaving. As shown in Figure 5, the pseudo-coded 2-ACK method is given for the 2-ACK packet sender side (N3) and the observing node side (N1) with the formal way of representing the 2-ACK execution process.

4. Digital Signature

In the conventional signature scheme, a handwritten signature is embodied with the documents which specify that this person is responsible for it. The importance of signature can be seen in everyday circumstances, such as contract signing, money withdrawn from the bank, letter writing, etc. One of the most identification and authentication mechanisms in a now day's digital world is the digital signature. It is a process to sign a message that is stored in electronic form, and then this signed message can be sent to the network towards its destination. It allows source users to create a code for the message that acts as a signature. A digital signature for any message can be created in the public key setup by taking a message hash value and encrypting it or signing it using a private key of its own. Basically, digital signature guarantees the integrity of the message and signer's identity. The digital signature scheme mainly offers some set of security abilities that very hard to implement in any other way.

4.1 Needs of Digital Signature

In general, the message authentication defends two communicating parties from any other third party that is exchanging the message with each other. But still, it does not provide the protection between them against each other. There may be numerous forms of the dispute between two parties could that are as follows:

a. Receiving party (Bob) may create a different message and claim that it has come from source party (Alice). For this, Bob creates a message and attached an authentication code with this message by using a shared key, which was shared by Alice and Bob, previously.
b. After sending the message, later Alice can deny that he has sent messages to Bob. So, there is no way for Bob to prove that this message has in fact received by the Alice.

In the above both situations, it could be said that there is no complete trust between two communicating parties. Due to this reason, something more than authentication is required.

The best way to avoid the above problem could be the use of the digital signature. The analogous to digital signature is the handwritten signature. The digital signature must meet specific attributes:

- Able to verify the sender identification along with the time and date of signature.
- Able to authenticate the content of the information at the time of signature.

- If any disagreement exists than any other third party must be able to verify it.

4.2 Digital Signature Techniques:

Any digital signature technique includes two different components: one is the signing algorithm ($SIGN_K$) and the second one is the signature verification algorithm (SIGN_VERk), both should be the polynomial-time functions of any key that is from key-space. The first one will be kept secret and the second one will be publicly available. The formal definition or steps of the creation of the digital signature and its verification schemes is presented out in Table 3. Consider any two communicating parties say the sender is Alice and the receiver is Bob. Now, Alice may create the message (X) and encrypt this message or sign the message using Signature (S) that depends on his own private key (d). After receiving the signed message, Bob will verify or decrypt this signed or an encrypted message: Y=S(X) using Alice public key (e) that is available in the Public Key Directory (PKD). For a pair of the message and signature/signed message (X, Y), the verification algorithm reverts either true or false that depends on whether signature Y is valid or not for created message X.

Table 3. Formal Way to Create and Verify Digital Signature Technique

- ✓ Digital Signature Creation and Verification are a five-tuple scheme that could be represented by five different variables (M, S, K, S_A, and V_A).
- ✓ It must fulfill the following prerequisites:
  - M: Finite set of all possible messages.
  - S: Finite set of all possible signatures.
  - K: Finite possible key space, i.e. list of possible keys.
  - S_A: Signature algorithm space set of all possible signature algorithms.
  - V_A: Signature verification algorithm space set of all possible signature verification algorithms.

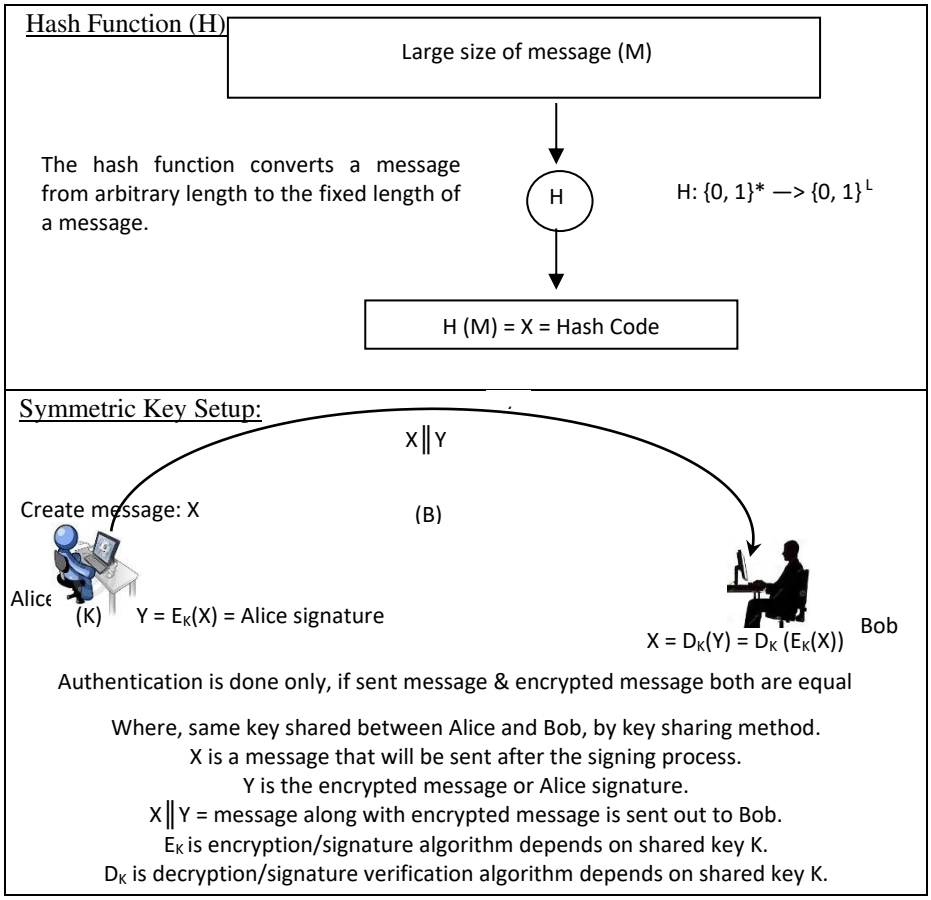
Algorithm: A Digital Signature Creation by Sender:

1. INPUT: k.
   Where, each key (k) belongs to key-space (K) (i.e. $k \in K$)
2. OUTPUT: S
   Where, S = Signature ($S \in S_A$)
3. Compute a *message digest* (m) of the message that is going to be sent
4. m lies between 1 and n-1,
   Where, n = modulus
5. Sender computes the $S = m^d \bmod n$ by using his own private key (d)
6. Return (S)
7. Sends this created signature (S) to receiver.

Algorithm: A Digital Signature Verification by Receiver:

1. INPUT: k.
   Where, each key (k) belongs to key-space (K) (i.e. $k \in K$)
2. OUTPUT: V
   Where, V = Verification $\in V_A$
3. Compute integer $V = S^e \bmod n$ by using sender public key (e).
4. Extracts the message digest (m) from this integer.
5. Separately computes the message digest (m') of the message which has been signed by sender.
6. If both are same i.e. m=m' that means signature is valid.
7. Return (True).

The hash function or hash code and possibilities of digital signature creation techniques are shown in Figure 6. A hash function (H) could be implemented in any size of the block of data that is variable length and generates a fixed message length as shown in Figure 6(A). A hash function is required because the implementation of the digital signature scheme on the large size of massage, especially in the public key setup is very costly. Figure 6(B) and 6(C) shows the creation of the digital signature and its verification in symmetric and public key setup, respectively. In symmetric key setup, Bob can play the role of the adversary by modifying the original content of a message. Alice does not have any way to prove his actual message. So, overall, these issues can be avoided by the public key setup. However, in both public and symmetric key setup, the only authentication can be made still confidentiality of information is not preserved. Authentication of the users as well as the confidentiality of information both could be maintained from digital creation schemes of Figure 6(D) and 6(E) because here the message is not directly sent. In both schemes, the signing process is done with Alice's private key. Finally, it is sent out in the channel using a symmetric shared key and Bob public key, respectively. In the state-of-the-art, there is a few digital signature schemes such as RSA, El-Gamal, Rabin algorithm, etc. Here, the RSA digital signature algorithm has been used.



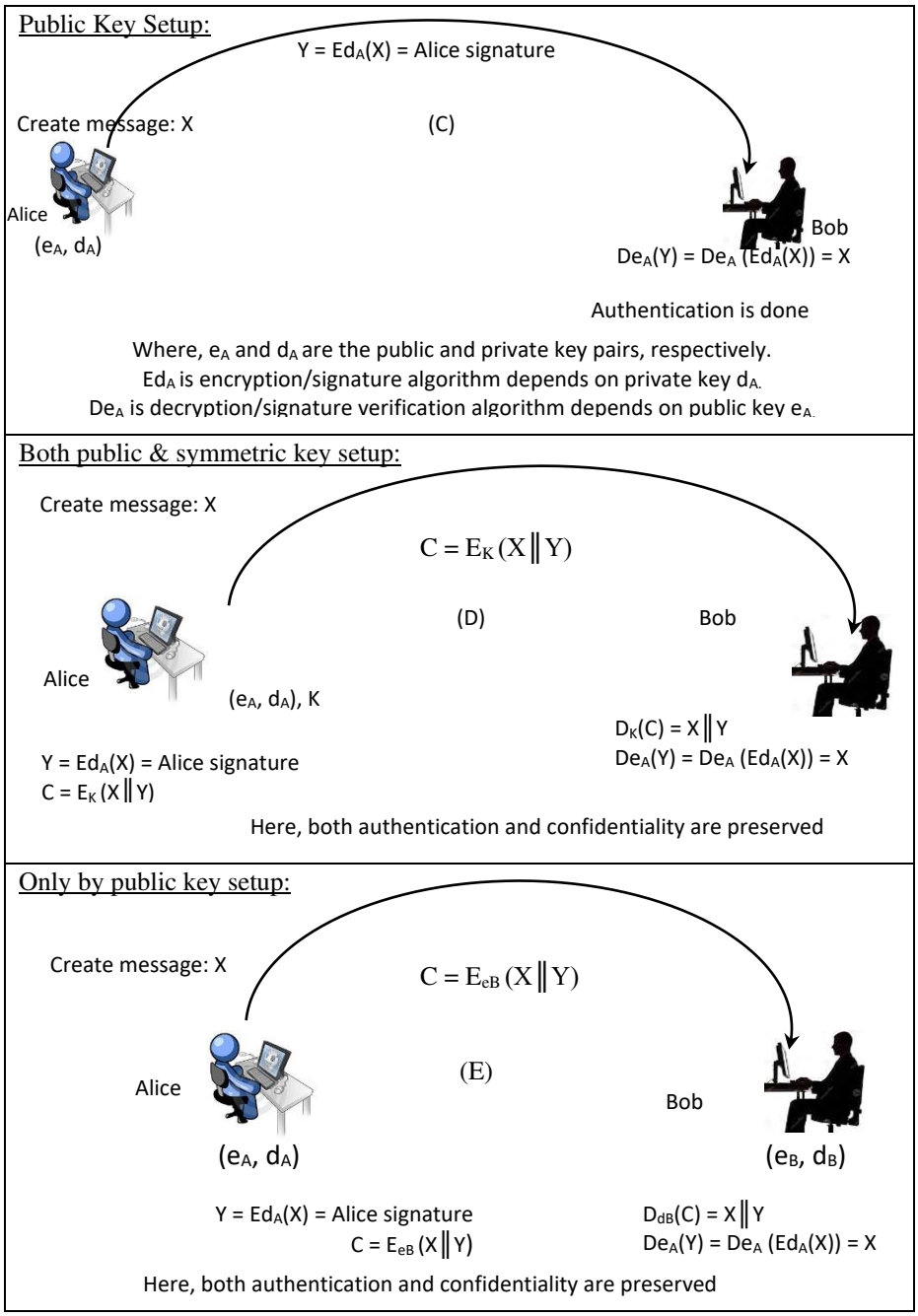


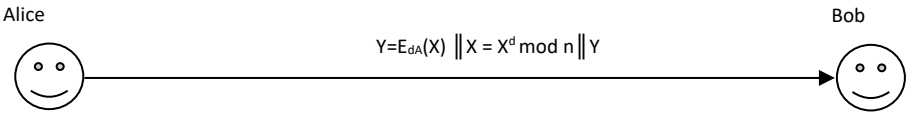
Figure 6. 6(A) Hash function and Digital Signature: 6(B) Symmetric Key Setup, 6(C) Public Key Setup, 6(D) both Public and Symmetric Key Setup, 6(E) Only by Public Key Setup

4.2.1: RSA Digital Signature Scheme

The Rivest, Shamir, and Adelman (RSA) cryptosystem can be used to provide a digital signature, and it is known as the RSA digital signature scheme. The required setup to create the RSA signature is demonstrated in Table 4. Moreover, Figure 7 illustrates the explicit demonstration of the creation and verification of the RSA digital signature scheme. RSA algorithm [44] is helpful to provide secure data transmission in a public-key cryptosystem that basically deals with digital signature including the message recovery scheme. The key generation in RSA digital signature is similar to the key generation in RSA.

Here, plaintext/all possible message space is M, So, (Message) X ∈ M

Keyspace (K): $\{(n, p, q, e, d)\} | ed \cong 1 \pmod{\Phi(n)}$
Public Key of Alice (e_A): (n, e)
Private/Secret Key of Alice (d_A): (p, q, d)



Key pairs: (e_A, d_A)

Now, Alice creates digital signature or encrypt this message "x" by using his own secret key (d_A):

$$E_{dA}(X) = X^d \bmod n$$

Then, digital signature along with message will be pumped into the network towards Bob.

Now, Bob verify this digital/signed signature or decrypt the message by using Alice public key (e_A) that is publically available in PKD:

$$De_A(Y = E_{dA}(X)) = Y^e \bmod n$$
$$= X^{ed} \bmod n \quad (Y = X^d \bmod n)$$
$$= X^{K\Phi(n)+1} \bmod n \quad (\because ed = K\Phi(n)+1)$$
$$= (X^{\Phi(n)} \bmod n)^K . X$$
$$= X$$

(By Euler's phi function, $X^{\Phi(n)} \bmod n = 1$, if $m \perp n$)

Where, $\perp$= relatively co-prime to each other.

If, send message X and decrypted message X both are same then verification is done.

According to Table (4), RSA signature is also five–tuple scheme that could be represented by five different variable as mention in Table. Therefore, RSA digital signature and its verification could also be presented as per following equations:

$$SIGN_K(X) = X^d \bmod n \quad \ldots (1)$$
$$SIGN\_VER_k(X, Y) = \text{True, if } X = Y^e \bmod n$$
$$= \text{False, if } X \neq Y^e \bmod n \quad \ldots (2)$$

Fig. 7. RSA Digital Signature: Creation and Verification

Table 4. In RSA Setup Key Generation Phase

Algorithm: A Key Generation Phase for RSA Setup to Create Digital Signature
1. INPUT: K. Where, K = required modulus bit length, 2. OUTPUT: An RSA Key pairs [e_A = (n, e), d_A = (p, q, d)]: Where, e_A = public key. d_A = private key. n = modulus that is the product of two large prime numbers p & q (n=p*q) not exceeding K bits in length, it should be one-way function (I.e. factorization must be hard). e = public exponent, a number less than and coprime to Φ(n) such that [gcd (e, Φ(n))=1], Where, Φ(n) is Euler's phi or totient function which value is equal to (p-1)(q-1). Required to ensure that "e" has inverse under mod Φ(n). (i.e. e^{-1}= d). (Here, inverse of "e" will exist because [gcd(e, Φ(n))=1]. d = private exponent such that ed ≅ 1 (modΦ(n)) [≅=congruent to]. 3. Choose value of "e" such that [gcd (e, Φ(n)) = 1]. 4. Repeat p ← genprime (K/2) Until (p mod e) ≠ 1 5. Repeat q ← genprime (K - K/2) Until (q mod e) ≠ 1 6. Compute n = p*q 7. Compute Φ(n) = (p-1)*(q-1) 8. Compute d = modinv(e, Φ(n)) = e^{-1} modΦ(n) 9. Return (n, e, d)

5. Problem Definition

The approach proposed is designed to solve three shortcomings of the Watchdog system, namely: receiver collision, limited transmission power, and false identity problem. In the case of receiver collisions (Figure 8), after I transmit Packet 1 to J, it will try to overhear whether J will forward this packet to K; meanwhile, X is forwarding Packet 2 to K. In such case, I overhear that J has successfully forwarded Packet 1 to K but failed to detect that K did not receive this packet due to a collision between Packet 1 and Packet 2 at K.

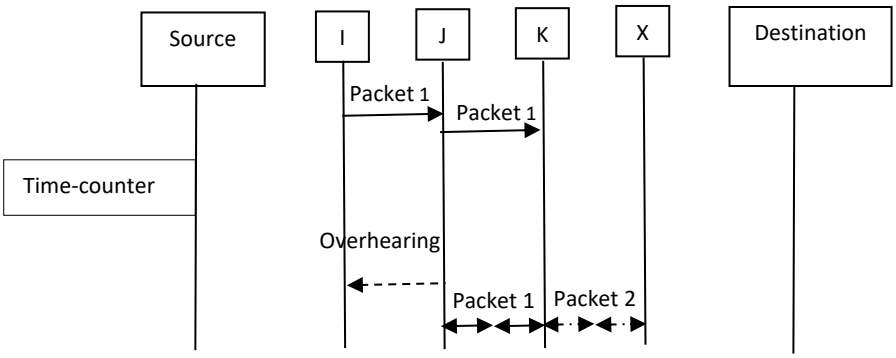


Fig. 8. Receiver Collisions

In the case of limited transmission power (Figure 9), J purposely decreases its transmission capacity to maintain its own battery life, so it is loud enough to be grasped by I, but still not strong enough to be heard by K.

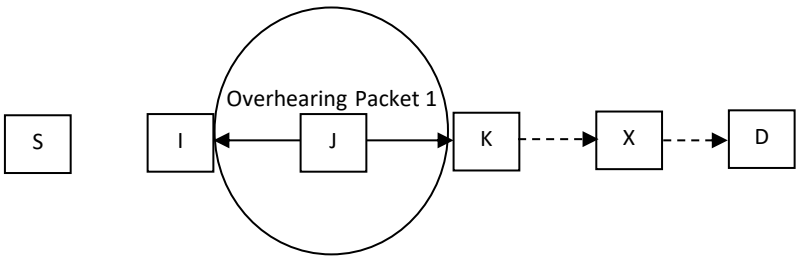


Fig. 9. Limited Transmission Power

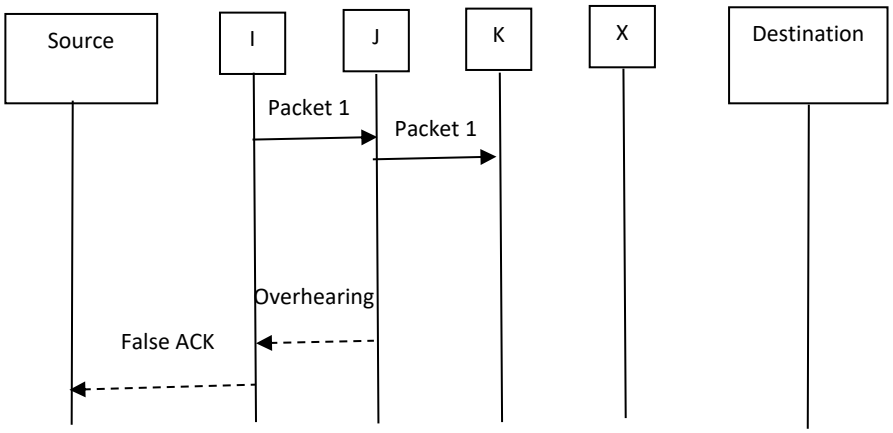


Fig. 10. False Misbehaviour Report

In the case of false misbehavior acknowledge (Figure 10), while I secretly recorded successfully that J forwarded Packet 1 to K, I also inform J as behaving badly. Due to the versatile platform and remote distribution of MANETs, attackers can easily catch and hack nodes to execute this attack to report misbehavior.

6. Proposed Method: DSSAM

DSSAM stands for a digitally signed secure acknowledgment method that using the digital signature technique to avoid the falsification of packets by the attacker. DSSAM consists of explicitly three major activities:

A. Secure ACK,
B. Node authentication,
C. Packet authentication

It uses the advantage of a 2-ACK method which already helpful to get overcome basic problems with Watchdog approaches, namely insufficient transmitting capacity and collision with the receiver. After that, we tried to solve the false misbehavior activity by securing acknowledgment, node authentication, and packet authentication. The function of such detection schemes largely depends on the acknowledgment packets. Hence, it is also very important to guarantee that acknowledgment packets are valid and authentic as well as secure. To this concern, a digital signature is introduced.

We safeguard two-layered defense for security. Additional bits allocated in the first layer are used to carry sequence numbers, keeping transmission time fixed to define the packets sequence in the proper interval for that time. This is done for the transmission of both packet and acknowledgment. The next layer is defined by twofold safeguarding the forwarded packets, by putting digital signature. According to the draft of DSR [45, 46], seven bits are reserved in the DSR header. These seven bits have been used to maintain sequence numbers. We assume bi-directional communication links with source and destination not being malicious. Both data packets and packets of acknowledgments must be digitally signed by the source and authenticated by the destination. In our proposed scheme, RSA is used to encrypt the packet.

7. Performance Evaluation

This section discusses the simulation method, setting up of simulations and review of comparative results with existing ones such as DSR, Watchdog, and 2-Ack.

7.1 Simulation Approach

To examine the performance of DSSAM with several kinds of attacks, we have planned two case scenarios to simulate diverse kinds of attacks by seeding proportionate misbehaving nodes in our simulation terrain setup:

CASE 1: Firstly, we conducted a packet-dropping and delay attack [47]. The malicious nodes lose all the packets got, meaning that mollify all the packets are lost. This scenario's concept is to measure the efficiency of intrusion detection against both the two limitations of the Watchdog; restricted transmission power and collision with the receiver as when there is a fixed range specified transmission power.

CASE 2: It is considered to examine intrusion detection systems performances against fake acknowledgment. Here, malicious nodes more cleverly behave with often falling the packets and return a fake acknowledgment whenever possible.

7.2 Simulation Setup

We have conducted the simulation using Intel Core i5 2.5 GHz processor and 8 GB 1600 MHz DDR3 main memory, with the consideration of both the physical layer and MAC layer 802.11b for simulation. Further, the experiment is performed through the QualNet Simulator-7.0 on a desktop as a simulation resource. For each scheme, each simulation ran 10 Telnet sessions and calculated the average. The 2-ACK scheme observational time is fixed at Tobs = 0.9 seconds. Unless otherwise stated the Rack = 0.25 recognition ratio being used by the 2-ACK scheme, acknowledgment miss ratio Rmis = 0.80 and a timeout value of T = 0.12 second. Along with the above explained parameters, there is Table 5, which gives the configuration of the experimental setup that is used for the analysis of the simulation. Thereafter, the performance of the proposed method has been evaluated with respect to by seeding malicious node percentage in terrain as 10%, 20%, 30%, 40%, 45% in uniformly distribution one by one.

Table 5. Parameters for Simulation

Parameters	
Packet Size	512 Bytes
Packet Rate	4 packets/sec
Data Traffic	CBR(UDP)
Dimensions	1000m x 1000m
Number of Nodes	50
Minimum Speed	1m/s
Maximum Speed	10m/s
Maximum Hops	5
Radio Transmission Range	200m
Simulation Time	1500s

CBR rate	50Kbps
Malicious node percentage in terrain	10%, 20%, 30%, 40%, 45%
Antenna Model	Omni-direction
Propagation Model	Two ray
Mobility Model	Random Waypoint
Channel Type	Wireless
Network Interface Type	PHY IEEE802.11 / Wireless
MAC Type	802.11b
Interface Queue Type	Queue/ Drop Tail / PriQueue
Link Layer Type	LL

We have observed the performance of DSSAM and compared it with Watchdog and 2-Ack. For this we have considered Packet Delivery Fraction (PDF), Routing Overhead (RO) and Average End-to-End Delay, as the performance metrics:

PDF is the proportion of the number of packets received by the top layer sources (i.e. application layer) and the number of packets obtained by the destination. This explains the rate of loss the transport protocol should experience.

PDF = (Data Packets Received)/ (Data Packets Sent)

RO: RO defines as the routing data of the network obtained by an application using a proportion of the required bandwidth. This additional data is called as routing overhead.

Average End-to-End Delay: It is the average amount of time that is taken by a packet to reach final destination from source. It is the sum of delays at links. The delay at a link is the sum of the following components (if, retransmission is not considered).

a. Processing delay
b. Queuing delay
c. Transmission delay
d. Propagation delay

Average End-to-End Delay = Σ (tr - ts)/Pr, where ts is the packet send time and tr is the packet receive time.

During the simulation, the origin node sends an RREQ packet to all other neighbors that broadcast will be within its range of communication. Neighbors received this RREQ message, so each neighbor adds their addresses consequently to the message and then sends an attached message to their neighbors. There is one important scenario that whenever any node receives more than one same RREQ, it completely denies it. In case any failed node is noticed, a message RERR is sent to the origin node, which usually implies a split link in flat routing protocols like DSR. When the RREQ destination node identified as the end destination node, this node activates an RREP message

and transfers back from the original RREQ message to the source node using the reverse route request process.

With reference to the digital signature system, we took up an open-source library called Botan [48]. For RSA schemes, we have considered a 512-b RSA key for each node in this network. For each node, we presumed that a private key and a public key were created and circulated in advance. The key file sizes of 512-b are 256 and 512 B, respectively. The signature file size for RSA is 120 B.

7.3 Results Analysis and Discussion

Case 1: Here, malicious nodes lost packets completely which passing through it. Figure 11 and Table 6 shows the results, based on packet delivery fraction. Here, we spot that all acknowledgment-based intrusion detection systems method like 2-ACK and DSSAM perform better than the Watchdog method. Our proposed method DSSAM outperforms Watchdog's performance by an average of 15% as 20% malicious nodes availability into the network. We observe that 2-ACK and DSSAM acknowledgment-based schemes are capable of detecting malfeasance with a receiver collision and limited transmission capacity. Nevertheless, if the percentage of malicious nodes exceeds 40%, the efficiency of our suggested DSSAM method is average 17% good than others. The reason behind that is the introduction of Packet Authentication Scheme (PAS) under DSSAM approach with choked route avoiding system, which is followed for the next time transmission through similar route with similar choked nodes. As whenever sender wait too long to receive a PAS acknowledgment from the destination node; means, that the waiting time exceeds the predefined threshold. This level is met for DSSAM only up to 50% of the involvement of malicious nodes, as network rises with more than 50% of malicious nodes, preceded by its fully compromised network. Thereafter, it again started decreasing because of generalize rule for any communication network; if malicious node presence increased by more than 50% then communication network system breaking up rate is increase by two times of normal decay rate in every 10% slot.

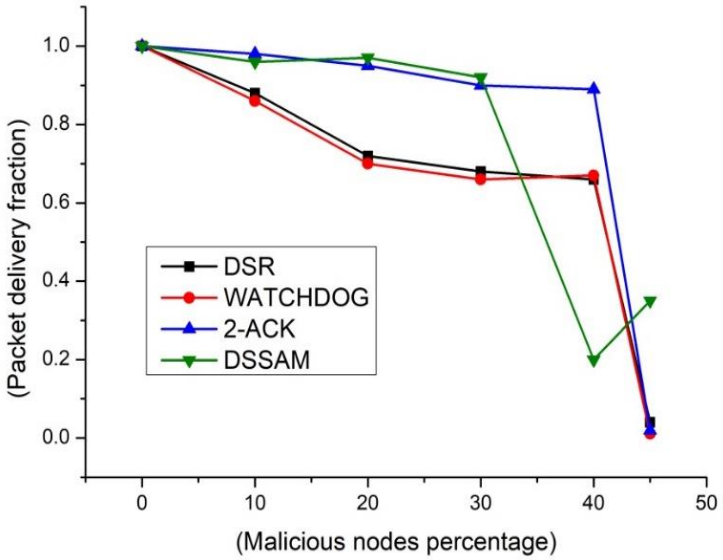


Fig. 11. Case 1 - Packet Delivery Fraction

The obtained routing overhead in case 1 of simulation environment is shown in Figure 12 and Table 6. It is observed that DSR and Watchdog scheme attains better result because they do not require acknowledgment method to detect mischief-nodes. As remaining two schemes; 2-ACK and DSSAM have effective overhead. Although, the DSSAM requires a digital signature and acknowledgment for all data packets, which cause to increase the routing overhead. Nevertheless, DSSAM still performs well compared to other acknowledgment techniques in most cases.

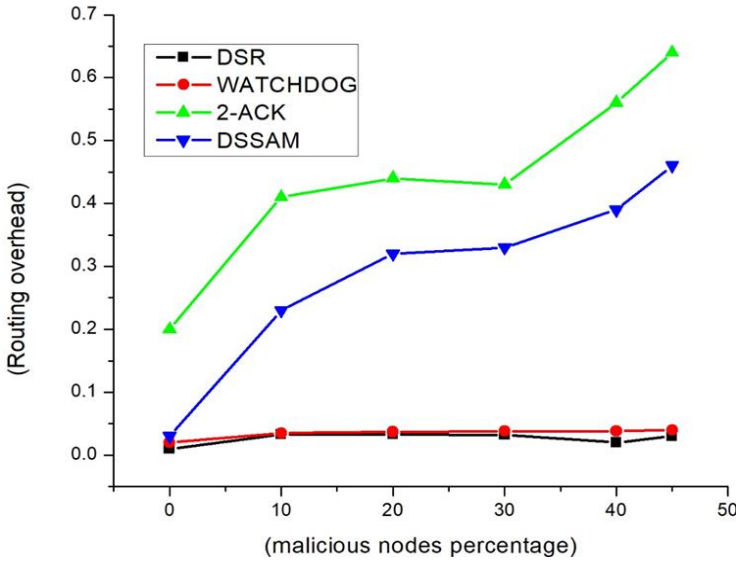


Fig. 12. Case 1 - Routing Overhead

The graph of average end-to-end delay for case 1 has been shown in Figure 13 and its value is tabulated in the Table 6. It is noticed that DSR and Watchdog method achieves better performance in terms of delay due to not requirement of acknowledgment packet to identify mischief-nodes as well in compare to 2-ACK and DSSAM. DSSAM took more average end-to-end delay time because of the enhanced feature of 2-ACK as digital signature incorporate for advance security feature as compare to previously existed method. However, if the percentage of malicious nodes exceeds by

30%, our suggested DSSAM framework is become bit quite slower more than others. Even watchdog performance is better in respect to average end-to-end delay.

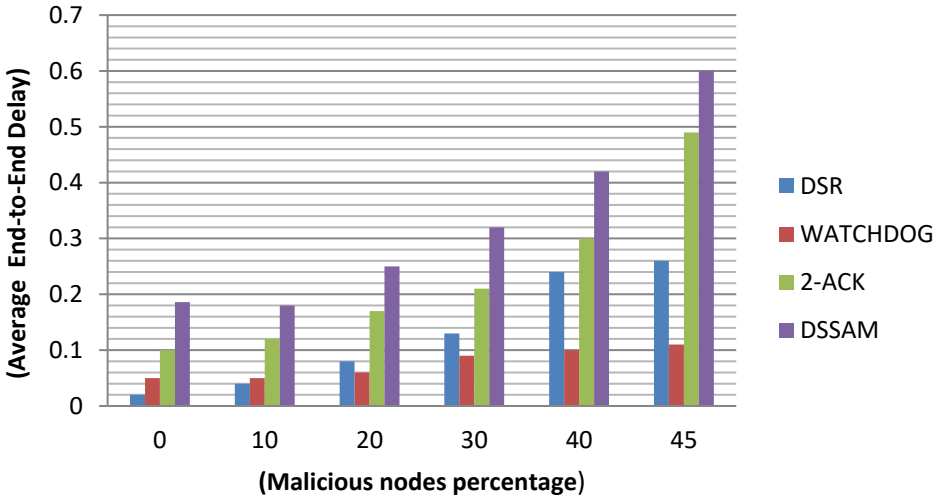


Fig. 13. Case 1 - Average End-to-End Delay

Case 2: Here, we seeded malicious nodes that send the fake acknowledgment to the source node as it is likely. This case is designed to check the intrusion detection system's performance under fake acknowledgment. Figure 14 and Table 6 show the results for packet delivery fraction. If the percentage of malicious nodes is 10%, DSSAM's output is around 3% higher than 2-ACK. DSSAM scheme beats all other schemes when the malicious nodes reach at 20% and 30%. DSSAM maintains the PDR to over 85% and if we compare it with 2-ACK than the output is 18% higher approximately. It performs similar to 2-ACK also in few point for particular this case. We be certain that the introduction of PAS scheme under DSSAM method framework mainly contributes to this performance.

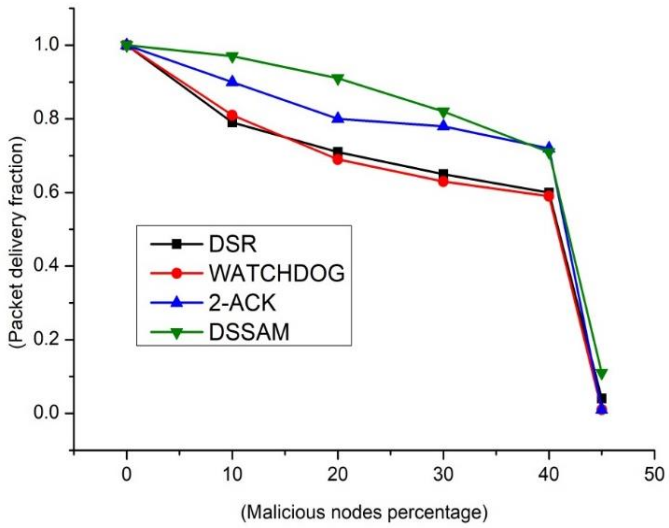


Fig. 14. Case 2- Packet Delivery Fraction

Figure 15 and Table 6 displays the simulation outcomes of the routing overhead in case 2. DSSAM in certain cases retains a lower overhead network particularly in comparison to 2-ACK and

Watchdog schemes. Routing overhead, however, is increasingly growing with the rise in malicious nodes. Therefore, there is a requirement for more digital signatures and acknowledgment packets. The routing overhead for DSSAM is more compared to other techniques, this is due to the hybrid nature and extra processing for digital signature. However, it is compensated by high packet delivery fraction and better security level in the packet communication.

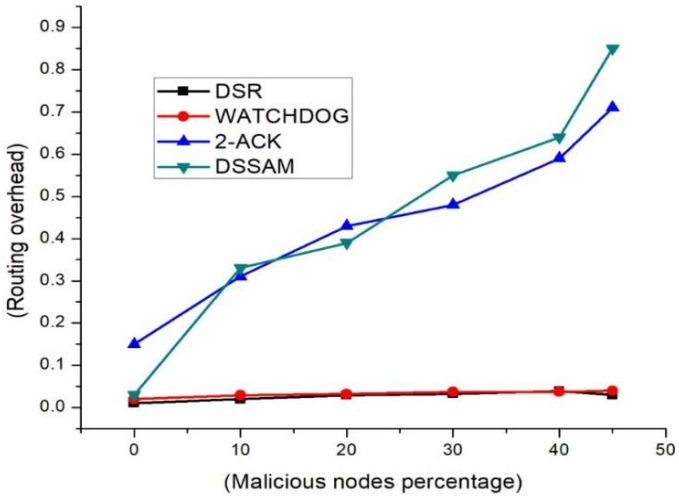


Fig. 15. Case 2- Routing Overhead

The outcome of average end-to-end delay for case 2 in Figure 16 and Table 6 exhibits moderate high for 2-ACK and DSSAM method with 40% or more malicious node presence. This high average delay presence due to the features of 2-ACK algorithm that incurred extra overhead for two hops moment with acknowledgement based handshaking property. In case 2, if more than 30% nodes shall start falsifying acknowledgement, in that point actual successful transmission would be reduced because proportionate percentage of retransmission increased to get actual transmission due to the false acknowledgement. DSSAM method is also facing same situation as 2-ACK but due to hybrid nature of extra security decrease the average end-to-end delay for DSSAM.

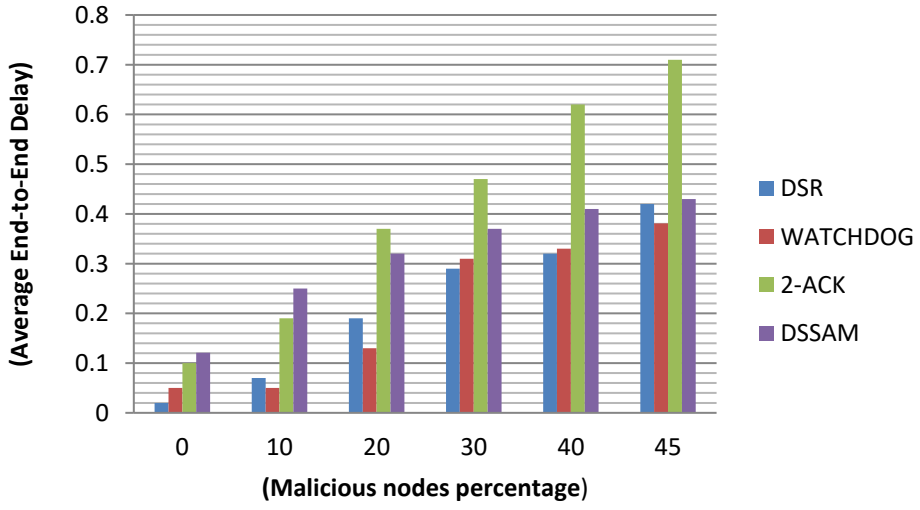


Fig. 16. Case 2- Average End-to-End Delay

Table 6. Average Results Outline

Case 1: Packet Delivery Fraction				
MALICIOUS NODES (%)	DSR	WATCHDOG	2-ACK	DSSAM
0%	1	1	1	1
10%	0.88	0.86	0.98	0.96
20%	0.72	0.70	0.95	0.97
30%	0.68	0.66	0.90	0.92
40%	0.66	0.67	0.89	0.20
45%	0.04	0.011	0.02	0.35
Case 1: Routing Overhead				
MALICIOUS NODES (%)	DSR	WATCHDOG	2-ACK	DSSAM
0%	0.01	0.02	0.2	0.03
10%	0.033	0.035	0.41	0.23
20%	0.033	0.037	0.44	0.32
30%	0.032	0.038	0.42	0.33
40%	0.02	0.038	0.56	0.39
45%	0.03	0.04	0.64	0.46
Case 1: Average End-to-End Delay				
MALICIOUS NODES (%)	DSR	WATCHDOG	2-ACK	DSSAM
0%	0.02	0.05	0.1	0.186
10%	0.04	0.05	0.12	0.18
20%	0.08	0.06	0.17	0.25
30%	0.13	0.09	0.21	0.32
40%	0.24	0.11	0.3	0.42
45%	0.267	0.128	0.49	0.6
Case 2: Packet Delivery Fraction				
MALICIOUS NODES (%)	DSR	WATCHDOG	2-ACK	DSSAM
0%	1	1	1	1
10%	0.79	0.81	0.90	0.97
20%	0.71	0.69	0.80	0.91
30%	0.65	0.63	0.78	0.82
40%	0.60	0.59	0.72	0.71
45%	0.04	0.01	0.01	0.11
Case 2: Routing Overhead				
MALICIOUS NODES (%)	DSR	WATCHDOG	2-ACK	DSSAM
0%	0.01	0.02	0.15	0.03
10%	0.020	0.029	0.31	0.33
20%	0.029	0.032	0.43	0.39
30%	0.0321	0.037	0.48	0.55
40%	0.039	0.037	0.59	0.64
45%	0.03	0.04	0.71	0.85
Case 2: Average End-to-End Delay				

MALICIOUS NODES (%)	DSR	WATCHDOG	2-ACK	DSSAM
0%	0.02	0.05	0.1	0.121
10%	0.07	0.06	0.19	0.25
20%	0.19	0.134	0.37	0.32
30%	0.29	0.31	0.47	0.37
40%	0.32	0.33	0.62	0.41
45%	0.42	0.381	0.71	0.43

7.4 Results Summary

The results revealed affirmative performances against Watchdog and 2-ACK, in the circumstances of receiver collision, limited transmission power, and false acknowledgement; proposed method also provides secure ACK with node authentication and packet authentication. Our proposed method DSSAM outperforms with Watchdog's and 2-ACK's performance in packet delivery fraction in both the cases for up to 50% malicious node presence in the communication network. In routing overhead concern, the non-acknowledgement methods attain better results than acknowledgement based method. Our proposed method also lagging here but it still performs well compared to other acknowledgment techniques in most cases. The average end-to-end delay in case 1, it is observed that DSR and Watchdog method achieves better result because they do not require acknowledgment method to detect mischief-nodes as compare to 2-ACK and DSSAM. DSSAM took more average end-to-end delay time because of the digital signature incorporate for advance security feature as compare to previously existed method.

8. Conclusion and Future Scope

There are many possible reasons for packet drop in MANETs that fall broadly under two types namely, intentional and unintentional mischief. The unintentional misbehavior could be caused by overloaded node (due to extreme dearth of CPU cycles and restricted buffer space), collision and traffic delays. The packet drop can happen due to connection errors because of intrusion or evaporation by the mischievous intruders. The packet-dropping attack represents a massive risk to secure the MANETs. This paper explains that we have described and simulated the method DSSAM in a standard environment and compared it with existing methods under different scenarios. The obtained simulation outcome provides enhanced performance against Watchdog and 2-ACK in the points of false misbehavior acknowledgment, collision with the receiver and the limited transmission capacity. We incorporated the digital signature in the method. While in a few circumstances, it creates more routing overhead, but increases the network's efficiency in terms of the fraction of packet transmission. It would be an interesting topic for a future research study to understand and estimate the performance when partially misbehaving nodes intentionally degrade performance

owing to their greediness for saving their own battery power. And to estimate the battery consumption with varying percentage of greedy nodes in the same environment.

List of Abbreviations

MANETs: Mobile Ad-hoc Networks

DSSAM: Digitally Signed Secure Acknowledgement Method

DSR: Dynamic Source Routing

WSN: Wireless Sensor Network

TARP: Trust Aware Routing Protocol

IDSs: Intrusion Detection System

2-ACK: Two hop acknowledgement method

SG: Security Goals

SA: Security Attack

DoS: Denial of Services

Cpkts**:** Counter of forwarded data packets

Cmis: Misbehaviour counter

$SIGN_K$: Signing algorithm

UA: Usability Attributes

SIGN_VERk: Signature verification algorithm

PKD: Public Key Directory

Tobs: Observation period of the 2-Ack scheme

Rack: Acknowledgement ratio

Rmis: Acknowledgement miss ratio

PDF: Packet Delivery Fraction

RO: Routing Overhead

Availability of data and materials: Not applicable.

Competing interests: The authors declare that they have no competing interests.

Funding: Not applicable.

Authors' contributions: Ashutosh Srivastava, Sachin Kumar Gupta are the main authors of the current paper. They contributed to the development of the ideas, design of the study, theory, result analysis, and paper writing. Mohd Najim, Nitesh Sahu, Geetika Aggarwal, and Bireshwar Dass Mazumdar contributed to the result analysis and paper revision. All authors read and approved the final manuscript.

References

[1] Internet Engineering Task Force, "*MANET Working Group Charter,"* Available from: IETF MANET Group Character Sector, (2013). https://tools.ietf.org/html/draft-ietf-manet-term [Last Access: 13 January 2020].

[2] B. Wu, J. Chen, J. Wu, and M. Cardei, "*A Survey of Attacks and Countermeasures in MANET,"* Wireless Network Security, Signals and Communication Technology, Springer, Boston, MA, 103-135, (2007). https://doi.org/10.1007/978-0-387-33112-6_5

[3] S. Tanwar, J. Vora, S. Tyagi, N. Kumar, and M. S Obaidat, "A systematic review on security issues in vehicular ad hoc network", Security and Privacy, Wiley, 1(5), (2018).

[4] J. Singh, and K. Singh, "Congestion control in vehicular ad hoc network: A review", Next-Generation Networks, Springer, 489-496, (2018).

[5] K. Kumar, S. Kumar, O. Kaiwartya, P. K. Kashyap, J. Lloret, and H. Song, "Drone Assisted Flying Ad-Hoc Networks: Mobility and Service-Oriented Modeling using Neuro-Fuzzy", Ad Hoc Networks, Elsevier, 106, (2020), 102242, 2020.6.22

[6] H. Miranda, and L. Rodrigues, *"Preventing Selfishness in Open Mobile Ad-hoc Networks*," IEEE Proceeding Seventh CaberNet Radicals Workshop, 1-6, (October 2002).

[7] M. Faisal, S. Abbasa, and H. U. Rahman, "Identity attack detection system for 802.11-based ad hoc networks", EURASIP Journal on Wireless Communications and Networking, 128, (2018).

[8] L. M. Feeney, and M. Nilsson, *"Investigating the Energy Consumption of a Wireless Network Interface in an Ad-hoc Networking Environment,"* IEEE INFOCOM, Conference on Computer Communications, Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No.01CH37213), Anchorage, AK, USA, 1548-1557, (2001).

[9] A. U Makarfi, K. M Rabie, O. Kaiwartya, X. Li, and R. Kharel, "Physical layer security in vehicular networks with reconfigurable intelligent surfaces", (2019), arXiv preprint arXiv:1912.12183.

[10] L. Buttyan, and J. P. Hubaux, *"Security and Cooperation in Wireless Networks. A Graduate Text Book,"*Cambridge University Press, (2007). http://secowinet.epfl.ch/fulltext/SeCoWiNetV1.5.1.pdf

[11] S. Marti, T. Giuli, K. Lai, and M. Baker, *"Mitigating Routing Misbehavior in Mobile Ad-hoc Networks,"* 6th Annual International Conference on Mobile Computing and Networking, Boston, MA, USA, 255-265, (August 2000) https://doi.org/10.1145/345910.345955.

[12] K. N. Qureshi, A. H. Abdullah, O. Kaiwartya, S. Iqbal, R. A. Butt, and F. Bashir, "*A Dynamic Congestion Control Scheme for safety applications in vehicular ad hoc networks*", Computers & Electrical Engineering, Elsevier, 72, 774-788, 2018.

[13] L. Buttyan, and J. P. Hubaux, *"Enforcing Service Availability in Mobile Ad-Hoc WANs,"* IEEE First Annual Workshop on Mobile and Ad-hoc Networking and Computing (Cat. No.00EX444), Boston, MA, USA, USA, 87-96, (August 2000), doi:10.1109/MOBHOC.2000.869216.

[14] J. P. Hubaux, T. Gross, J. Y. LeBoudec, and M. Vetterli, *"Toward Self-Organized Mobile Ad-hoc Networks: The Terminodes Project,"* IEEE Communications Magazine, 118–124, 39(1), (2001). doi:10.1109/35.894385.

[15] O. Kaiwartya, S. Kumar, D. K. Lobiyal, A. H. Abdullah, and A. N. Hassan, *"Performance Improvement in Geographic Routing for Vehicular Ad hoc Networks*", Sensors, MDPI AG, Basel, Switzerland, 14(12), 22342-22371, (2014). DOI: https://doi.org/10.3390/s141222342

[16] M. Alotaibi, *"Security to wireless sensor networks against malicious attacks using Hamming residue method*", EURASIP Journal on Wireless Communications and Networking, 8, (2019).

[17] S. Buchegger, and J. Y. Le Boudec, *"Performance Analysis of the CONFIDANT Protocol: Cooperation of Nodes, Fairness in Dynamic Ad-Hoc Networks,"* 3rd ACM International Symposium on Mobile Ad-hoc Networking and Computing, Switzerland, 226-236, June 2002. https://doi.org/10.1145/513800.513828

[18] S. Zhong, J. Chen, and Y. R. Yang, *"Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad Hoc Networks,"* IEEE INFOCOM, San Francisco, USA, 1-11, (2003).

[19] O. Kaiwartya, and S. Kumar, "*Guaranteed Geocast Routing Protocol for Vehicular Adhoc Networks in Highway Traffic Environmen*t", Wireless Personal Communications, Springer US, 83(4), 2657-2682, (2015).

[20] M. Jakobsson, J. P. Hubaux, and L. Buttyan, "*A Micropayment Scheme Encouraging Collaboration in Multi-Hop Cellular Networks,"* Financial Cryptography, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, Vol 2742, 15-33, (January 2003). https://doi.org/10.1007/978-3-540-45126-6_2

[21] B. Chaudhary, and K. Singh, "*Pseudonym generation using genetic algorithm in vehicular ad hoc networks"*, Journal of Discrete Mathematical Sciences and Cryptography, Taylor & Francis, 22(4), 661-677, (2019).

[22] A. U Makarfi, K. M Rabie, O. Kaiwartya, K. Adhikari, X. Li, M. Quiroz-Castellanos, and R. Kharel, "*Reconfigurable Intelligent Surfaces-Enabled Vehicular Networks: A Physical Layer Security Perspective"*, (2020), arXiv preprint arXiv:2004.11288

[23] K. Liu, J. Deng, P. K. Varshney, and K. Balakrishnan, "*An Acknowledgment-Based Approach for the Detection of Routing Misbehavior in MANETs,"* IEEE Transaction on Mobile Computing, 6(5), 536-550, (2007).

[24] L. Abusalah, A. Khokhar, and M. Guizani, *"Trust Aware Routing in Mobile Ad-hoc Networks,"* IEEE GLOBECOM, Communications Society, San Francisco, CA, USA, 1-5, (December 2006). DOI: 10.1109/GLOCOM.2006.264.

[25] N. Soganile, T. Baletlwa, and B. Moyo, *"Hybrid Watchdog and Pathrater Algorithm for Improved Security in Mobile Ad-hoc Networks,"* International Conference on Wireless Networks, ICWN'15, 162-167, (July 2015).

[26] R. Murugan, and A. Shanmugam, *"Cluster Based Trust Mechanism for Mitigation of Internal Attacks in Mobile Ad-hoc Networks,"* International Journal of Soft Computing, 7(6), 294-301, (2012). DOI: 10.3923/ijscomp.2012.294.301.

[27] L. Zhou, and Z. Haas, *"Securing Ad-hoc Networks,"* IEEE Network Magazine, 13(6), 24-30, (1999).

[28] A. Singh, M. Maheshwari, and N. Kumar, *"Security and Trust Management in MANET,"* Information Technology and Mobile Communication, AIM 2011, Communications in Computer and Information Science, New York: Springer-Verlag. 147, 384-387, (2011). DOI: 10.1007/978-3-642-20573-6_67

[29] F. Daryabar, A. Dehghantanha, and H. Broujerdi, *"Investigation of Malware Defense and Detection Techniques,"* International Journal of Digital Information and Wireless Communications, 1(3), 645-650, 2011.

[30] T. Anantvalee, and J. Wu, "*A Survey on Intrusion Detection in Mobile Ad-hoc Networks,"* Wireless Network Security, Springer, Boston, MA, 159-180, (2008). https://doi.org/10.1007/978-0-387-33112-6_7

[31] Y. Zhang, W. Lee, and Y. Huang, *"Intrusion Detection Techniques for Mobile Wireless Networks,"* Mobile Networks and Applications, 1-16, (2003).

[32] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, "*A Survey of Intrusion Detection Techniques in Cloud,*" Journal of Network and Computer Applications, 36(1), 42-57, (January 2013).

[33] S. Kumari, M. K. Khan, and M. Atiquzzaman, *"User Authentication Schemes for Wireless Sensor Networks: A Review,"* Ad-hoc Networks, 27, 159–194, (2015). https://doi.org/10.1016/j.adhoc.2014.11.018

[34] K. Wong, Y. Zheng, J. Cao, and S. Wang, *"A Dynamic User Authentication Scheme for Wireless Sensor Networks,"* International Conference on Sensor Networks, Ubiquitous, Trustworthy Computing, IEEE Computer Society, Taichung, Taiwan, 244-251, (June 2006). DOI: 10.1109/SUTC.2006.1636182

[35] B. Vaidya, D. Makrakis, and H. Mouftah, "*Two-Factor Mutual Authentication with Key Agreement in Wireless Sensor Networks,"* Security and Communication Networks, 9(2), 171–183, (2012), http://dx.doi.org/10.1002/sec.517.

[36] S. G. Yoo, K. Y. Park, and J. Kim, "*A Security-Performance-Balanced User Authentication Scheme for Wireless Sensor Networks,"* International Journal of Distributed Sensor Networks, 1-11, (2012). DOI: 10.1155/2012/382810.

[37] H. R. Tseng, R. H. Jan, and W. Yang, *"An Improved Dynamic User Authentication Scheme for Wireless Sensor Networks,"* IEEE Global Communications Conference, Washington, DC, USA, 985-990, (November 2007). DOI: 10.1109/GLOCOM.2007.190

[38] M. K. Khan, and K. Alghathbar, *"Cryptanalysis and Security Improvements of Two-Factor User Authentication in Wireless Sensor Networks,"* Sensors 10 (3), 2450–2459, (2010). DOI: 10.3390/s100302450.

[39] S. Athmani, A. Bilami, and D. E. Boubiche, "*EDAK: An Efficient Dynamic Authentication and Key Management Mechanism for Heterogeneous WSNs*," Future Generation Computer Systems, 92, 789-799, (2017). DOI:10.1016/j.future.2017.10.026

[40] P. Ballarini, L. Mokdad, and Q. Monnet, "*Modeling Tools for Detecting DoS Attacks in WSNS*," Security and Communication Networks, 6, 420–436, (2013). https://doi.org/10.1002/sec.630

[41] M. A. Ferrag, L. Maglaras, A. Argyriou, D. Kosmanos, and H. Janicke, "*Security For 4G And 5G Cellular Networks: A Survey of Existing Authentication and Privacy-Preserving Schemes*," Journal of Network and Computer Applications, 101, 55-82, (2018). https://doi.org/10.1016/j.jnca.2017.10.017

[42] T. Sheltami, A. Al-Roubaiey, E. Shakshuki, and A. Mahmoud, *"Video Transmission Enhancement in Presence of Misbehaving Nodes in MANETs,"* International Journal on Multimedia System, 15(5), 273-282, (2009).

[43] A. Patwardhan, J. Parker, A. Joshi, M. Iorga, and T. Karygiannis, "*Secure Routing and Intrusion Detection in Ad-hoc Networks,"* 3rd International Conference on Pervasive Computing Communication, 191-199, (August 2005).

[44] R. Rivest, A. Shamir, and L. Adleman, *"A Method for Obtaining Digital Signatures and Public-Key Cryptosystems,"* Communications of the ACM, 21(2), 120-126, (1978).

[45] D. Johnson, and D. Maltz, *"Dynamic Source Routing in Ad-hoc Wireless Networks,"* Mobile Computing. Norwell, MA: Kluwer, chapter 5, 153–181, (1996).

[46] D. B. Johnson, D. A. Maltz, and J. Broch, "*DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad-hoc Networks*," Ad-hoc Networking, edited by Charles E. Perkins, Chapter 5, Addison-Wesley, 139-172, (2001).

[47] S. Om, and M. Talib, *"Wireless Ad-hoc Network under Black-hole Attack,"* International Journal of Digital Information and Wireless Communications, Society of Digital Information and Wireless Communications 1(3), 591-596, (2011).

[48] Botan: *"Crypto and TLS for Modern C++ Library,"* Online: http://botan.randombit.net/. [Last Access: October 2019].

Figures

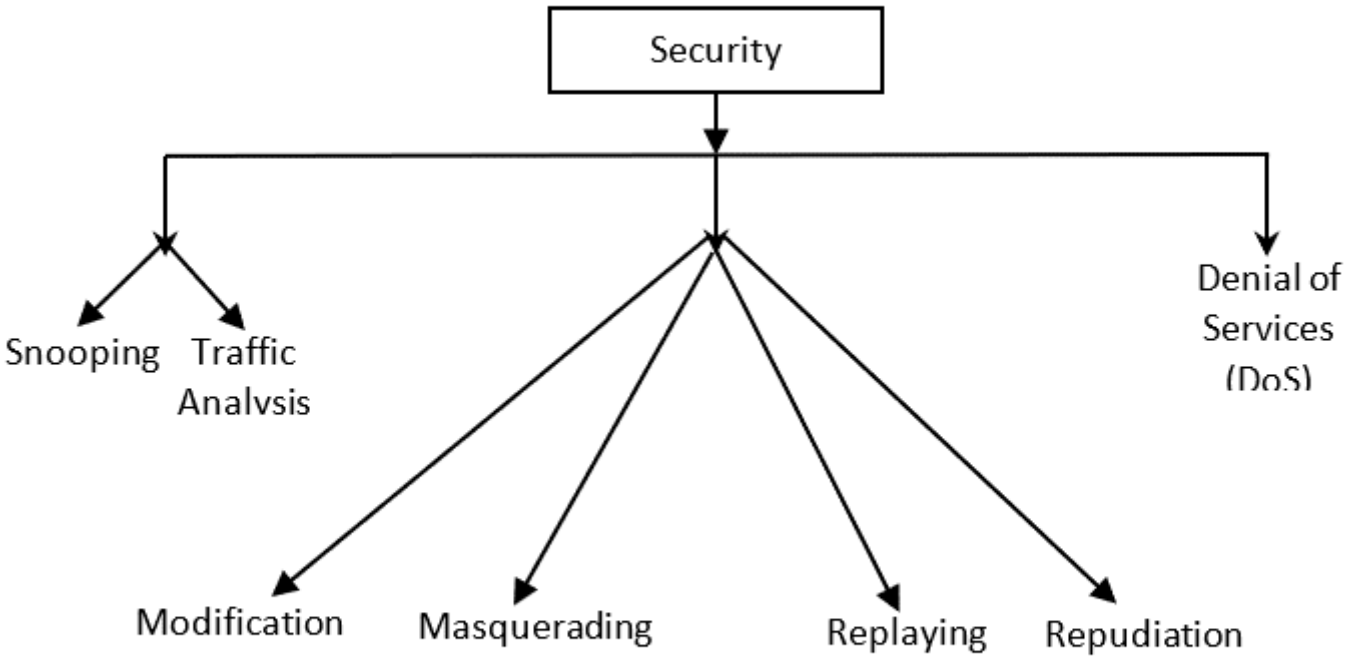


Fig. 1. Various Possible Security Attacks

Figure 1

Various Possible Security Attacks

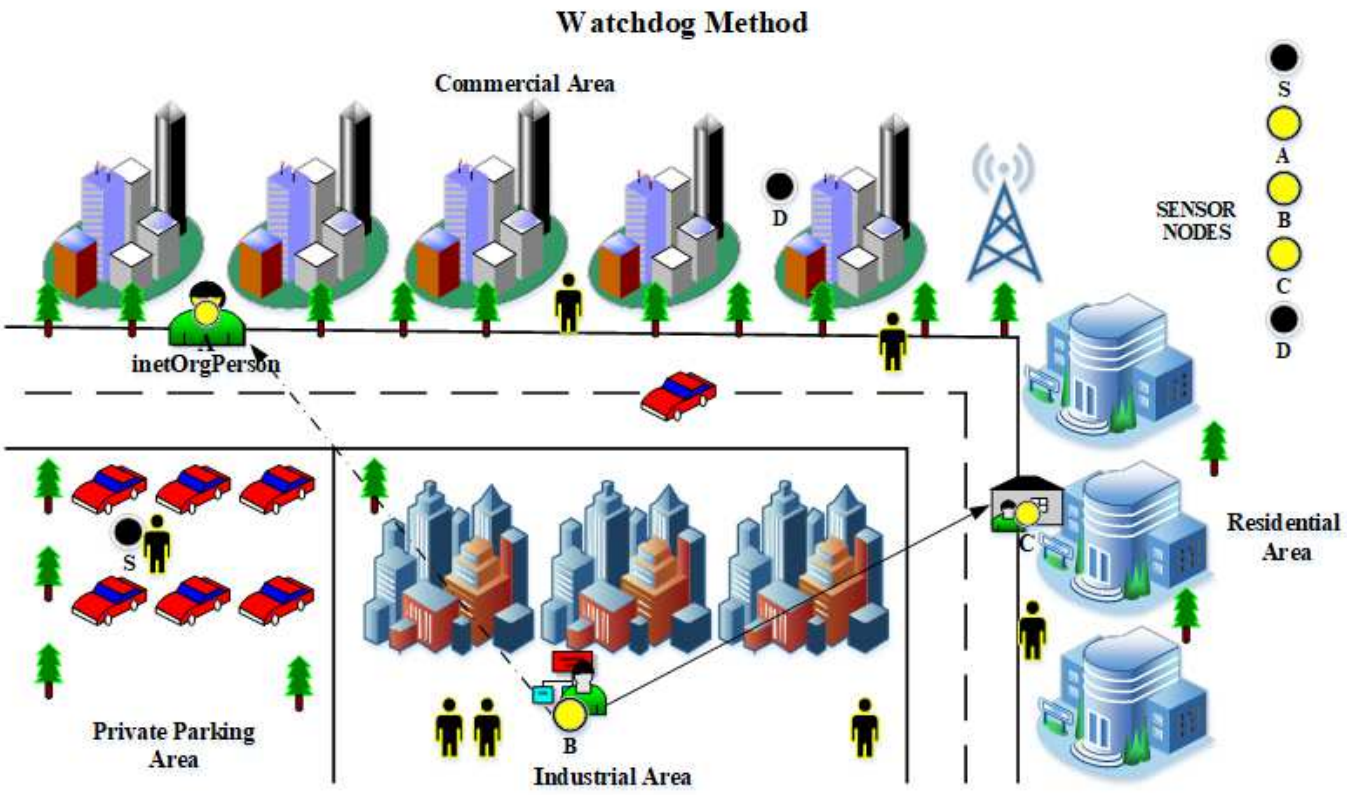


Fig. 2. Watchdog Method

Figure 2

Watchdog Method

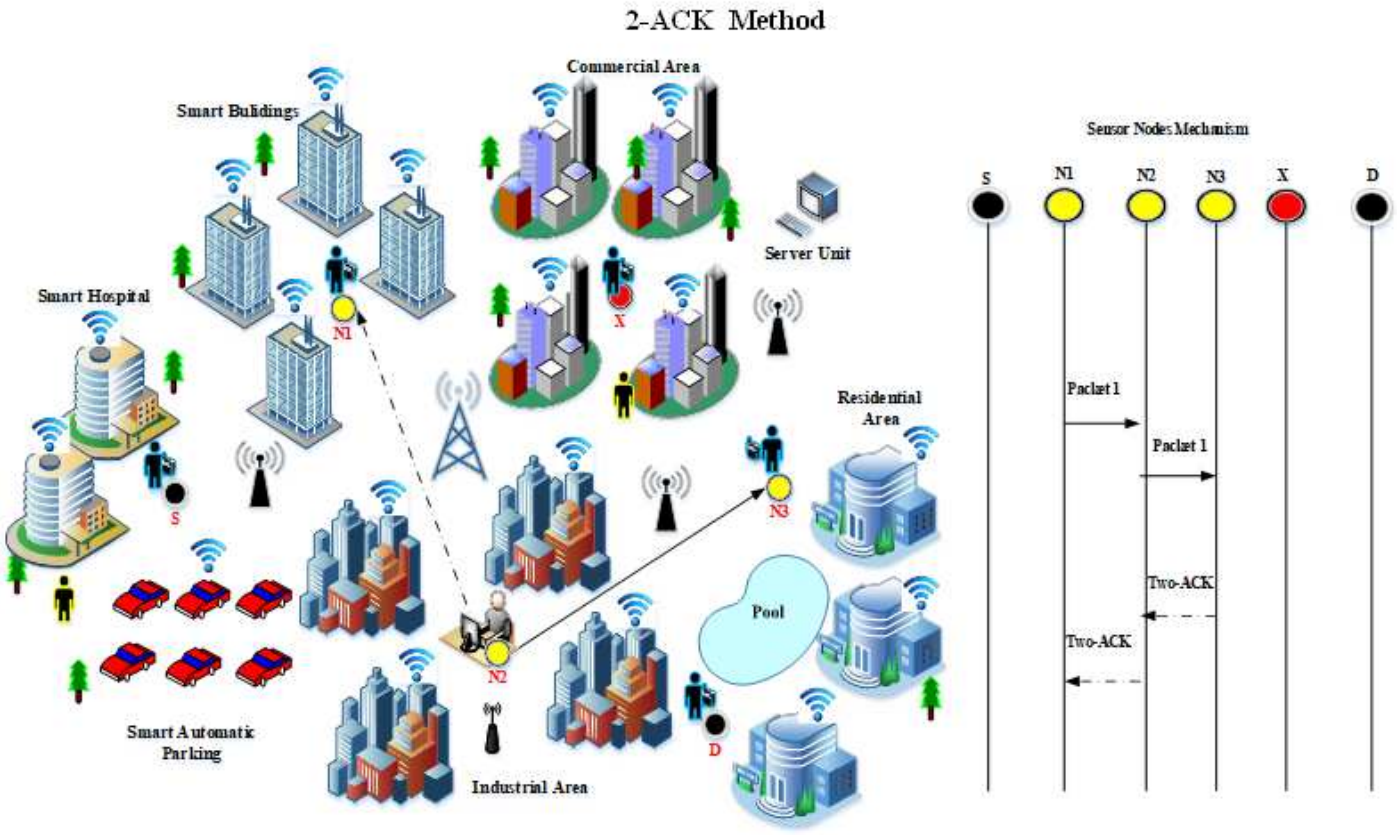


Fig. 3. 2- ACK Method

Figure 3

2-ACK Method

N2 Next Hop Receiver	N3 Second-Hop Receiver	Cpkts Packets Transmitted	Cmis 2-ACK packets Missed	LIST List of data Packet IDs

Fig. 4. Data Structure Maintain by Observing Node

Figure 4

Data Structure Maintain by Observing Node

Pseudo code for 2-ACK Method

We use the triplet $N1 \rightarrow N2 \rightarrow N3$ in Figure 2 as an example to illustrate 2-ACK's pseudo code. Note that such codes are run on each of the sender/receiver of the 2-ACK packets.

X.1 2-ACK Packet Sender Side (Node $N3$)

2: $Cpkts \leftarrow 0$, $Cack \leftarrow 0$, $i \leftarrow n$ # *Initialization at node N3*
3: **while** *true* **do**
4: **if** (*data packet received*) **then**
5: $Cpkts$ ++ # *Increase the counter of received packets*
6: **if** ($Cack = Cpkts < Rack$) **then** # *The data packet needs to be acknowledged*
7: *prepare MAC*
8: *prepare* 2-ACK *with ID*
9: *send* 2-ACK
10: $Cack$ ++, i -- # *Increase the counter of acknowledged packets*
11: **end**
12: **end**
13: **end**

X.2 Receiver (Observer) Side (Node $N1$)

Parallel process 1 (receiving *hn*)
1: **while** *true* **do**
2: **if** *receive hn from the* 2-ACK *packet sender* **then**
3: *record hn*, $i \leftarrow n$
4: **end**
5: **end**
Parallel process 2 (receiving 2-ACK packets)
6: **while** *true* **do**
7: *randomly select Tstart > current time* # *Start the observation*
8: **while** *current time < Tstart* **do**
9: # null
10: **end**
11: $LIST \leftarrow \varphi$, $Cpkts \leftarrow 0$, $Cmis \leftarrow 0$ # *Initialization at node N1*
12: **while** *current time < Tstart + Tobs* **do** # *Observation period is not expired*
13: **if** (*data packet forwarded*) **then**
14: $LIST \leftarrow LIST \cup data\ ID$ # *Add a data ID to LIST*
15: $Cpkts$ ++ # *Increase the counter offorwarded packets*
16: *setup timer* (τ) *for data ID* # *Record the time*
17: **end**
18: **if** (2-ACK *packet received*) **then**
19: *search data ID carried by* 2-ACK *from LIST*
20: **if** (*found*) **then** == *A* 2-ACK *packet for a data ID received*
21: *check validity of hi*
22: $LIST \leftarrow LIST - data\ ID$ # *Remove data ID from LIST*
23: *clear timer for ID*
24: **end**
25: **end**
26: **if** (*timeout event happens*) **then** # 2-ACK *packet for a data ID is not received*
27: $LIST \leftarrow LIST - data\ ID$ == *Remove data ID from LIST*
28: $Cmis$ ++ # *Increase misbehaviour counter*
29: **end**
30: **end**
31: **if** ($Cmis = Cpkts > Rmis$) *then* # *The observation period expires*
32: *send link misbehaviour report*
33: **end**
34: **end**

Fig. 5. 2-ACK Executions Process

Figure 5

2-ACK Executions Process

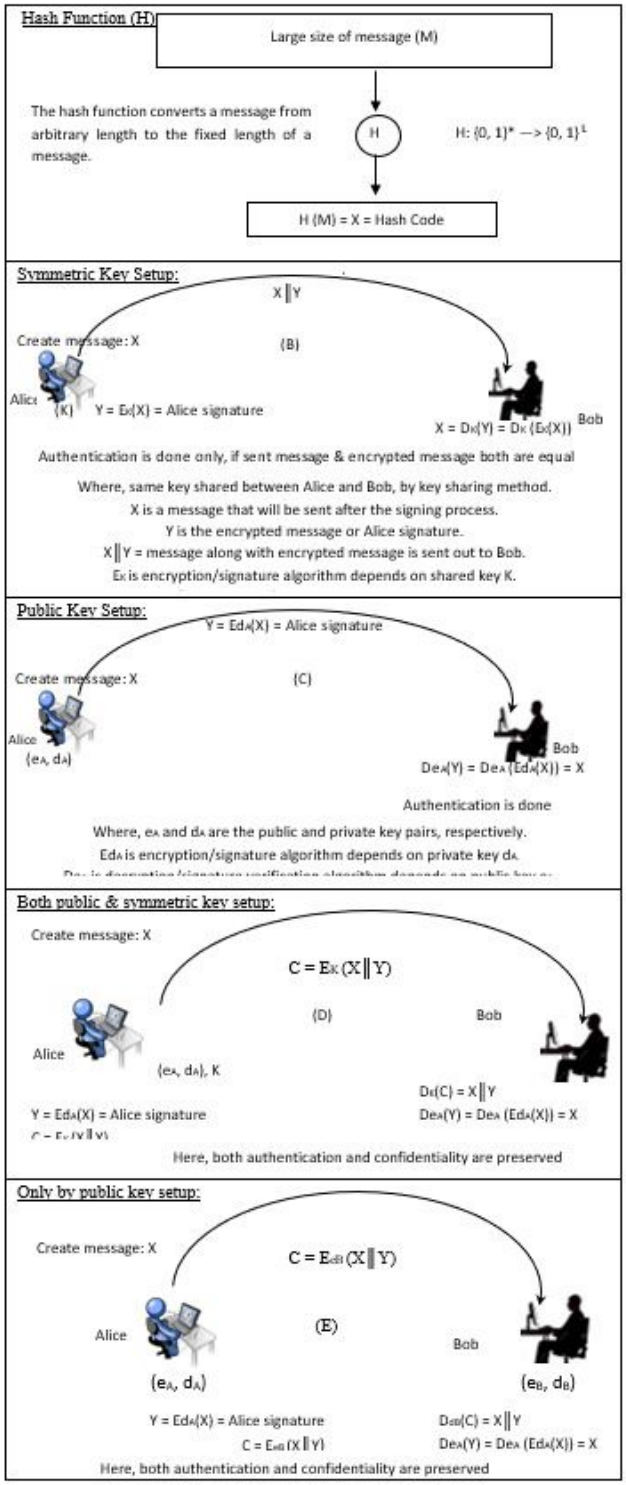


Figure 6. 6(A) Hash function and Digital Signature: 6(B) Symmetric Key Setup, 6(C) Public Key Setup, 6(D) both Public and Symmetric Key Setup, 6(E) Only by Public Key Setup

Figure 6

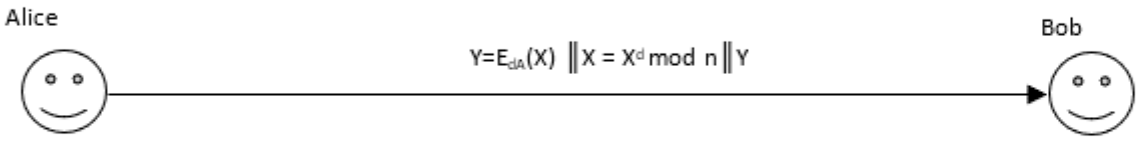
(A) Hash function and Digital Signature: 6(B) Symmetric Key Setup, 6(C) Public Key Setup, 6(D) both Public and Symmetric Key Setup, 6(E) Only by Public Key Setup

Keyspace (K): $\{(n, p, q, e, d)\} | ed \cong 1 \pmod{\Phi(n)}$

Here, plaintext/all possible message space is M, So, (Message) $X \in M$

Public Key of Alice (e_A): (n, e)

Private/Secret Key of Alice (d_A): (p, q, d)



Key pairs: (e_A, d_A)

Now, Alice creates digital signature or encrypt this message "x" by using his own secret key (d_A):

$$E_{dA}(X) = X^d \bmod n$$

Then, digital signature along with message will be pumped into the network towards Bob.

Now, Bob verify this digital/signed signature or decrypt the message by using Alice public key (e_A) that is publically available in PKD:

$$De_A(Y = E_{dA}(X)) = Y^e \bmod n$$
$$= X^{ed} \bmod n \quad (Y = X^d \bmod n)$$
$$= X^{K\Phi(n)+1} \bmod n \quad (\because ed = K\Phi(n)+1)$$
$$= (X^{\Phi(n)} \bmod n)^K . X$$
$$= X$$

(By Euler's phi function, $X^{\Phi(n)} \bmod n = 1$, if $m \perp n$)

Where, $\perp$= relatively co-prime to each other.

If, send message X and decrypted message X both are same then

According to Table (4), RSA signature is also five–tuple scheme that could be represented by five different variable as mention in Table. Therefore, RSA digital signature and its verification could also be presented as per following equations:

$$SIGN_K(X) = X^d \bmod n \ldots \quad (1)$$

$$SIGN\_VER_k(X, Y) = \begin{cases} True, & \text{if } X = Y^e \bmod n \\ False, & \text{if } X \neq Y^e \bmod n \end{cases} \quad \ldots \quad (2)$$

Fig. 7. RSA Digital Signature: Creation and Verification

Figure 7

RSA Digital Signature: Creation and Verification

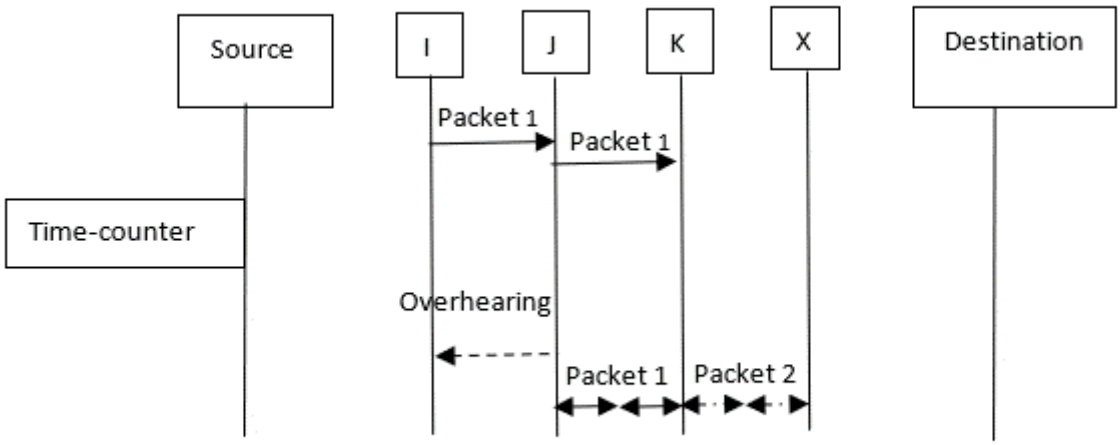


Fig. 8. Receiver Collisions

Figure 8

Receiver Collisions

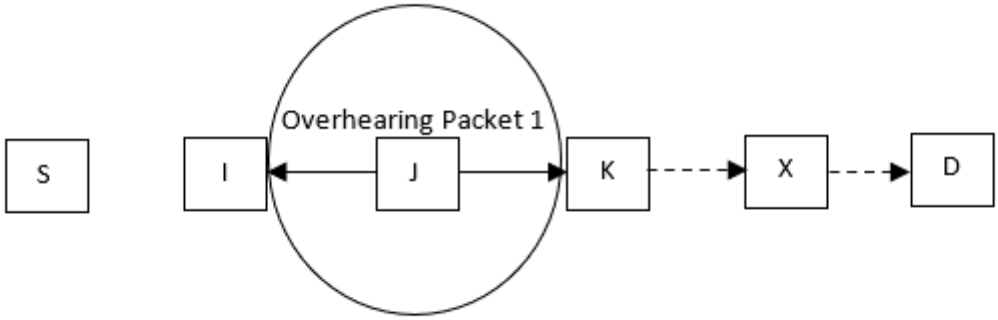


Fig. 9. Limited Transmission Power

Figure 9

Limited Transmission Power

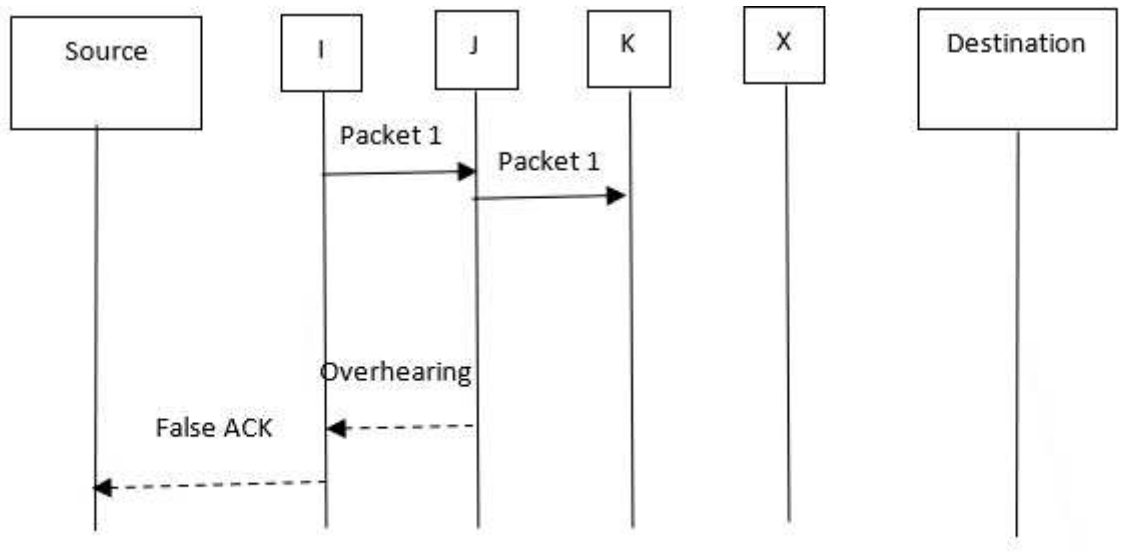


Fig. 10. False Misbehaviour Report

Figure 10

False Misbehaviour Report

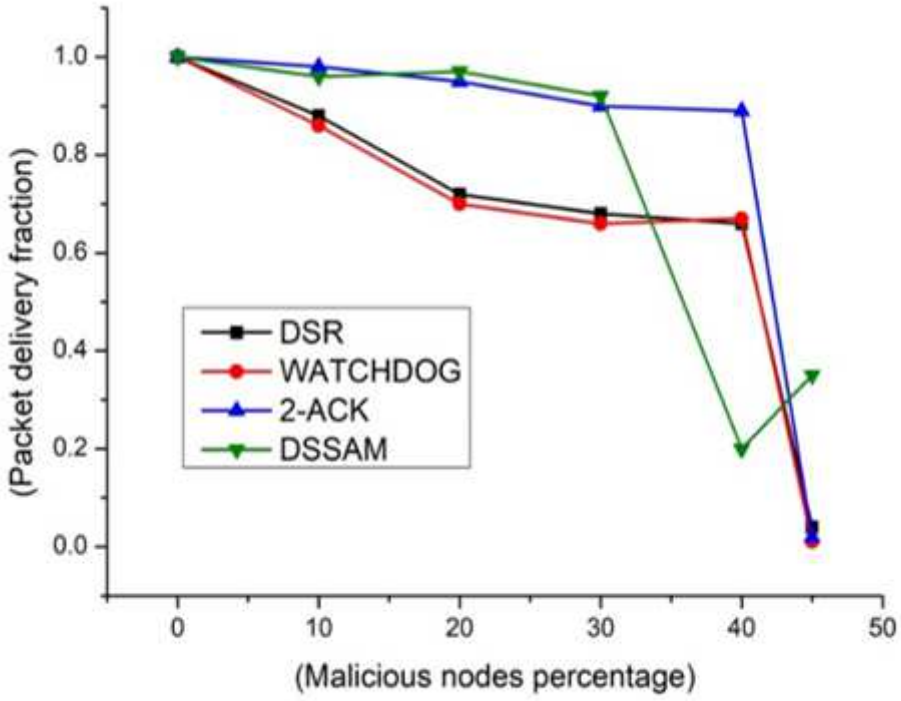


Fig. 11. Case 1 - Packet Delivery Fraction

Figure 11

Case 1 -Packet Delivery Fraction

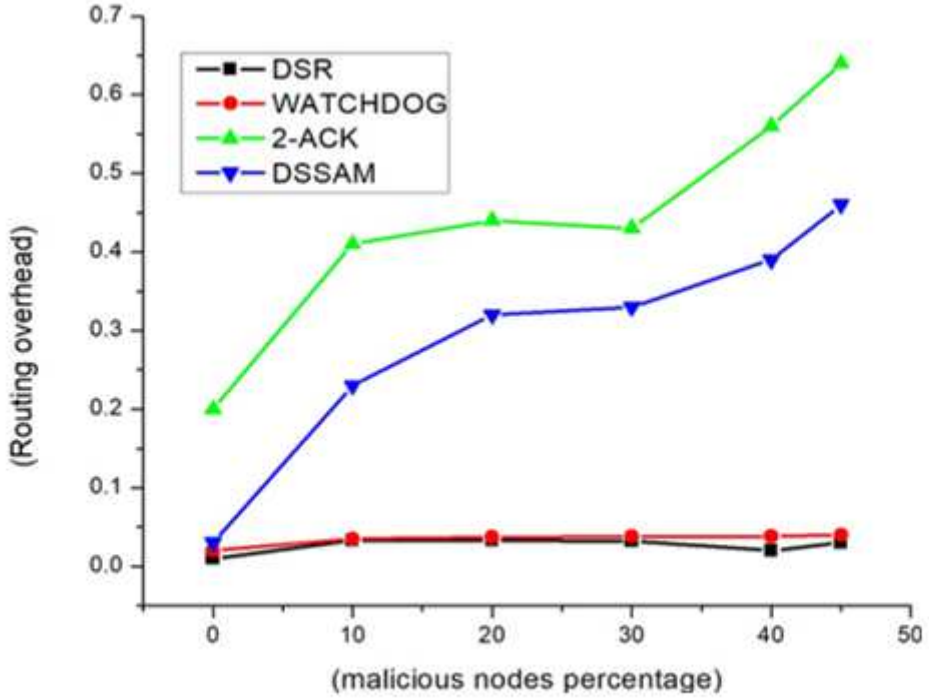


Fig. 12. Case 1 - Routing Overhead

Figure 12

Case 1- Routing Overhead

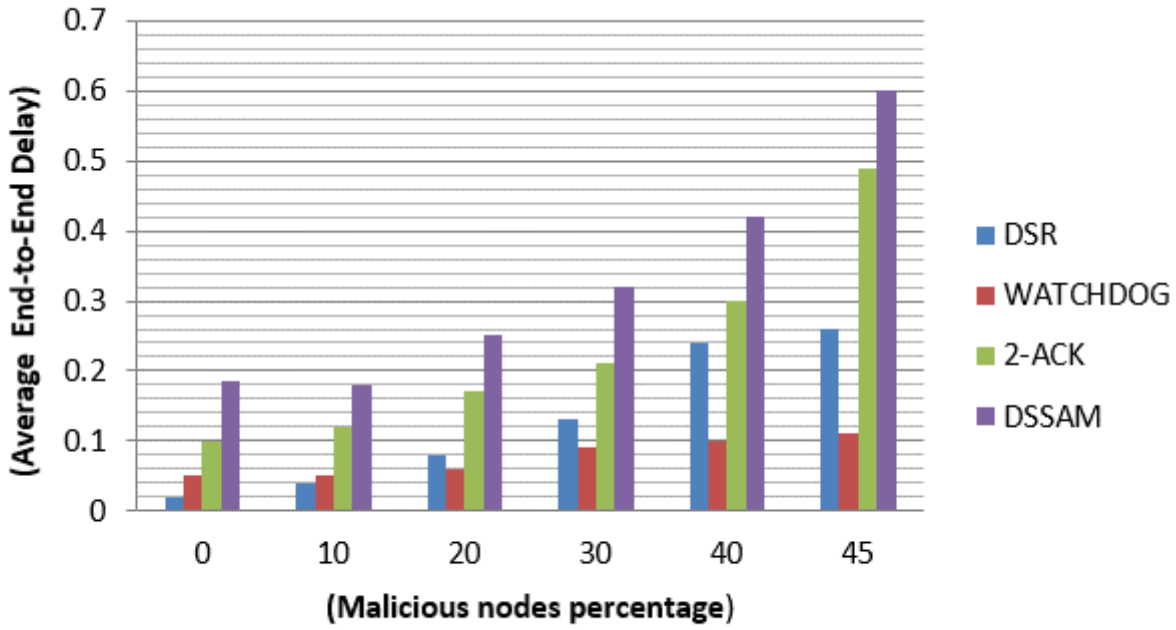


Fig. 13. Case 1 - Average End-to-End Delay

Figure 13

Case 1- Average End-to-End Delay

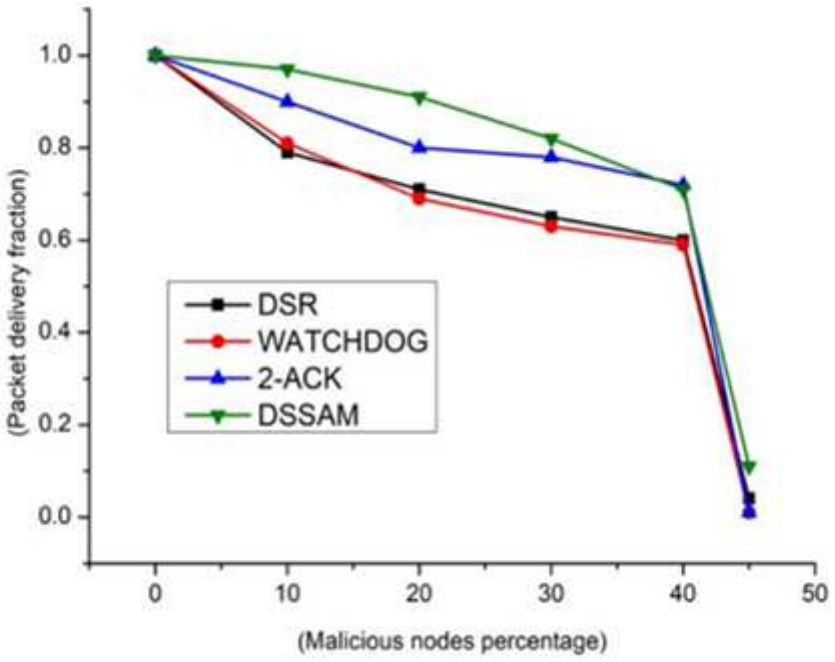


Fig. 14. Case 2- Packet Delivery Fraction

Figure 14

Case 2 -Packet Delivery Fraction

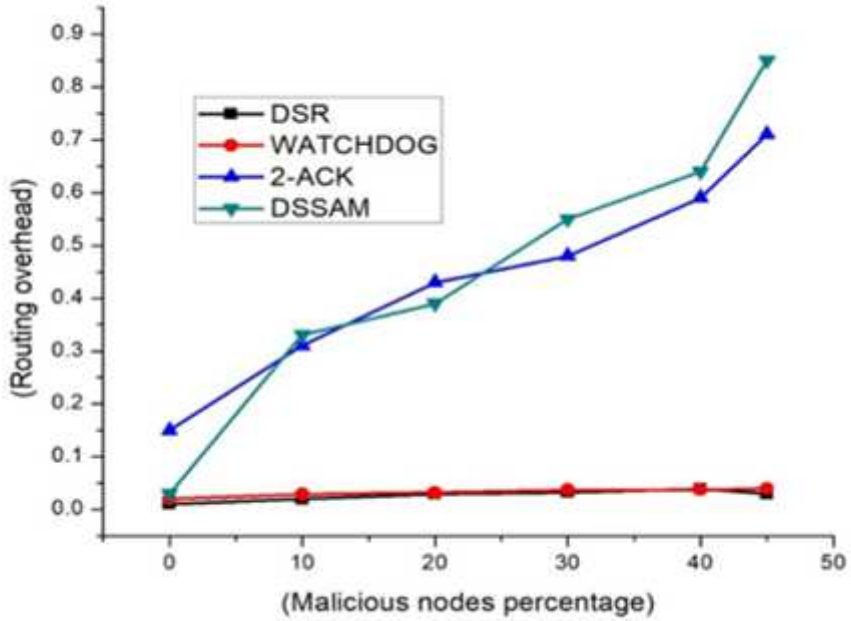


Fig. 15. Case 2- Routing Overhead

Figure 15

Case 2- Routing Overhead

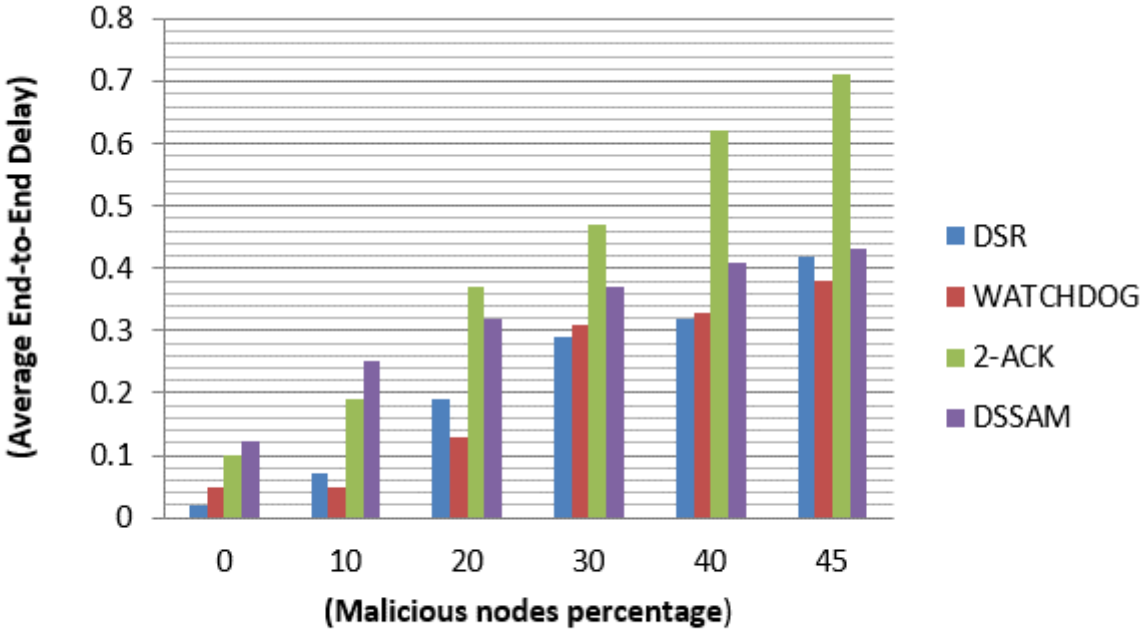


Fig. 16. Case 2- Average End-to-End Delay

Figure 16

Case 2- Average End-to-End Delay