International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 3 | ISSN : 2456-3307

Detecting Attacks in MANET using Secure Zone Routing Protocol

Ritu Aggarwal
M. Tech, Computer science & Engineering, Jagadhri, Haryana, India

ABSTRACT

MANET is a wireless network of mobile devices that has the ability to self-configure and self organize and it is characterized by an absence of centralized administration and network infrastructure. In this paper, present Zone Routing Protocol (ZRP); the most popular routing protocol. The importance of the proposed solution lies in the fact that it ensures security as needed by providing a comprehensive architecture of Secure Zone Routing Protocol (SZRP) based on efficient secure neighbor discovery, secure routing packets, detection of malicious nodes, and preventing these nodes from destroying the network. In order to fulfill these objectives, both efficient key management and secure neighbor mechanisms have been designed to be performed prior to the functioning of the protocol. Our approach is based on the Zone Routing Protocol (ZRP); the most popular hybrid routing protocol. The importance of the proposed solution lies in the fact that it ensures security as needed by providing a comprehensive architecture of Secure Zone Routing Protocol (SZRP) based on efficient key management, secure neighbor discovery, secure routing packets, detection of malicious nodes, and preventing these nodes from destroying the network. In order to fulfill these objectives, both efficient key management and secure neighbor mechanisms have been designed to be performed prior to the functioning of the protocol.

Keywords : Ad-Hoc Networks, Secure Routing, Secure Neighbor Discovery, Digital Signature, Zone Routing Protocol, Secure Zone Routing Protocol

I. INTRODUCTION

Mobile ad-hoc network is a wireless and baseless network which does not require any physical media or infrastructure to communicate between wireless ad-hoc network nodes. A mobile ad hoc network (MANET) is a self-configuring infrastructure less network of mobile devices which is connected by wireless.

This Wireless is a technology that allows users to access information and services in spite of the geographic position.

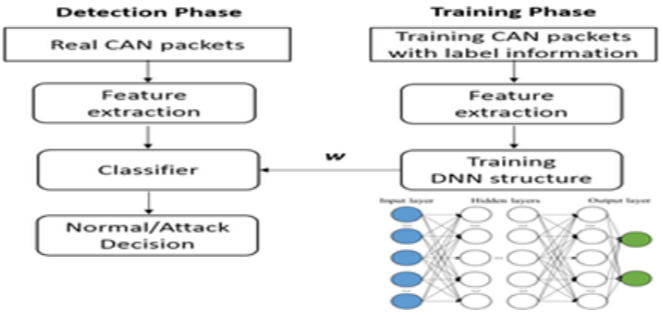

Figure 1. System Architecture

Mobile ad hoc network (MANET) is an autonomous group of mobile users who communicate with each other without any fixed infrastructure and centralized administration [2]. Since the hosts are mobile, the network topology may change rapidly and unpredictably over time. The attractive features of ad-hoc networks such as open medium,dynamic

topology,absence of central authorities,and distributed cooperation hold the adhoc networks across arrange of civil, scientific,military an industrial applications [1]. However,thes echaracteristicsmake ad-hocnetworks vulnerable to different types of attacks and make implementing security in adhoc network a challenging task.The main security problems that need to be dealt with inad-hoc networks include: the identity authentication of devices that wish to talk to each other, the secure key establishment of keys among authenticated devices,the secure routing in multi hop networks,and the secure transfer of data[12].This means that the receiver should be able to confirm that he identity of the source or the sender(i.e.,one hop previous node) is indeed who or what it claims to be.It also means that the receiver should be able to verify that the content to message has not been altered either maliciously or accidentally in transit In this paper, we propose securing one of the most popular hybrid protocols :zone routing protocol(ZRP). ZRP [16] aims to address excess bandwidth and long route request delay of proactive and reactive routing protocols. It combines the advantages of these approaches by maintaining an up-to-date topological map centred on each node. The separation of anode's local neighbourhood from the global topology of the entire network allows fo rapplying different approaches, and thus taking advantage of each technique's features for a given situation. These local neighbourhoods are called zones; each node may be within multiple overlapping zones, and each zone may be of a different size. The nodes of a zone are divided into peripheral nodes whose minimum distance to the centre is exactly equal to zone radius, gray nodes, and interior nodes whose minimum distance to the centre is less than zone radius, white nodes.Conventional ZRP isnot secure as it does not consider security requirements..First,we use an efficient key management mechanism that is considered as a prerequisite for any security mechanism. Then, we provide a secure neighbor detection scheme that relies on neighbor discovery,time and location based protocol.

II. RELATED WORKS

A mobile ad hoc network (MANET) is a wireless communication network which does not rely on a pre-existing infrastructure or any centralized management. Securing the exchanges in MANETs is compulsory to guarantee a widespread development of services for this kind of networks. The deployment of any security policy requires the definition of a trust model that defines who trusts who and how. Our work aims to provide a fully distributed trust model for mobile ad hoc networks.

In this paper, we propose a fully distributed public key certificate management system based on trust graphs and threshold cryptography. It permits users to issue public key certificates, and to perform authentication via certificates' chains without any centralized management or trusted authorities [1]. The trust is always present implicitly in the protocols based on cooperation, in particular, between the 26 entities involved in routing operations in Ad hoc networks. Indeed, as the wireless range of such nodes 27 is limited, the nodes mutually cooperate with their neighbors in order to extend the remote nodes and 28 the entire network. In our work, we are interested by trust as security solution for OLSR protocol. This fits particularly with characteristics of ad hoc networks [2]. A mobile ad hoc network (MANET) refers to a network designed for special applications for which it is difficult to use a backbone network. In MANETs, applications are mostly involved with sensitive and secret information. Since MANET assumes a trusted environment for routing, security is a major issue. In this paper we analyze the vulnerabilities of a pro-active routing protocol called optimized link state routing (OLSR) against a specific type of denial-of-service (DOS) attack called node isolation attack. Analyzing the attack, we propose a mechanism called enhanced OLSR (EOLSR) protocol which is a trust based technique to secure the OLSR

nodes against the attack [3]. Optimized Link State Routing is a routing protocol that has been extensively studied for mobile ad-hoc networks. Link spoofing, which disturbs the routing service, is one of the critical security problems related to the OLSR protocol. Existing approaches against link spoofing attack have several drawbacks. In this paper, propose an LT-OLSR protocol that broadcasts Hello messages to neighbors within two-hops to defend networks against link spoofing attacks [4]. Mobile Ad hoc network is consists of mobile nodes and can organize them self without requiring any infrastructure. Due to wireless communication any node can join or leave network which causes lot of security constraint and due to limited battery many researchers are doing researches on energy saving routing in MANET. In OLSR there is need of selecting MPR set, which minimize unnecessary.in network, that conserve energy of node in network [5]. Two measures to counter attacks against OLSR: prevention that solves some protocol's vulnerabilities and countermeasures that treat misbehavior and inconsistency concerned by the vulnerabilities that have not been solved with prevention measures. The resulting mechanisms allow resolving the OLSR vulnerabilities which are due to the easy usurpation of node's identity, and the lack of links verification at the neighborhood discovery [6]. Collusion Attack is an attack against Mobile Ad Hoc Networks and is based on Optimised Link State Routing (OLSR) Protocol. In this attack, two attacking nodes collude to prevent routes to a target node from being established in the network. Packet Delivery Ratio (PDR) of nodes 2-hops away from the victim drops to 0%. Multi Point Relay (MPR) selection process in OLSR is exploited to achieve route denial. In this paper, propose a novel attack resistant method named Forced MPR Switching OLSR (FMS-OLSR), in which, whenever a node observes symptoms of the attack, it temporarily blacklists potential attackers [7]. Mobile ad hoc networks (MANETs) are well known to be vulnerable to various attacks due to their lack of centralized control, and their dynamic topology and energy constrained operation. Much research in securing MANETs has focused on proposals which detect and prevent a specific kind of attack such as sleep deprivation, black hole, grey hole, rushing or sybil attacks. In this paper propose a generalized intrusion detection and prevention mechanism. We use a combination of anomaly-based and knowledge based intrusion detection to secure MANETs from a wide variety of attacks [8]. Mobile ad hoc networks are vulnerable to a variety of network layer attacks such as black hole, gray hole, sleep deprivation & rushing attacks. In this paper we present an intrusion detection & adaptive response mechanism for MANETs that detects a range of attacks and provides an effective response with low network degradation. We consider the deficiencies of a fixed response to an intrusion; and we overcome these deficiencies with a flexible response scheme that depends on the measured confidence in the attack, the severity of attack and the degradation in network performance [9]. However, MANETs are vulnerable to various attacks at all layers, including in particular the network layer, because the design of most MANET routing protocols assumes that there is no malicious intruder node in the network [10]. OLSR relies on the cooperation between network nodes, it is susceptible to a few colluding rogue nodes, and in some cases even a single malicious node can cause routing.

III. DESIGN OF SECURE ZONE ROUING PROTOCOL (SZRP):

For our design to be suitable for ad-hoc networks, the following design goals should be met:[19]

- Few computational steps to reserve the limited power of all ad-hoc devices since too many computational steps will drain the battery.
- Balanced protocol, which means that all nodes should perform approximately the same number of heavily computations.
- Few packets flow with small size since large packets are spitted into several packets to

match the available communication bandwidth where sending many packets contradicts with the previous design goal.

- Restricted number of heavy computations, such as modular exponentiations, to save battery power although the processors of most ad-hoc devices are becoming more powerful and can perform these computations. Network Assumptions
- The physical layer of a wireless network is often vulnerable to denial of service attacks such as jamming, and many researchers have proposed mechanisms to resist physical jamming such as spread spectrum [5]. So, this type of attack is beyond the scope of this paper.
- We assume that the network links are either unidirectional or bidirectional; that is, if node A is able to transmit to some node B, node B does not necessarily have the ability to transmit to node A. Most recent researches that have been proposed to secure routing protocols assume bidirectional links [6-10]; although this is not always true. Node Assumptions[20]
- We assume that all nodes have loosely synchronized clock, and have the ability to define their location in order to perform neighbor authentication. Accurate time synchronization and location can be maintained with Global Position System, GPS [11].
- We do not assume trusted hardware. Secure routing with trusted hardware is much simpler since node compromise is assumed to be impossible.
- We assume that nodes in ad-hoc networks are resource constrained. Thus, in IERP, we use efficient symmetric cryptography in hop-to-hop transfer, rather than relying on expensive asymmetric cryptographic operations. Especially on CPU-limited devices, symmetric cryptographic operations (such as hash functions) are three to four orders of magnitude faster than asymmetric cryptographic operation [6, 12, 13,14.18]. We assume that each node has its private/public key pair, and has the ability to know the public keys of all other nodes.
- We base our design on the absence of public key infrastructure, or any trusted distribution center. Most previous works on secure MANETs routing protocols rely on them for the secrecy and authenticity of keys stored in nodes. However, this requirement of a central trust authority and pre-configuration is neither practical nor feasible in MANETs due to self-deployment, dynamic topology, and the lack of central authorities.

A. Key Generation Key generation is the process of calculating new key pairs for security purposes. In our design, this includes generation of public/private key pair for digital signature. The generation process is performed when the node is created (bootstrapping phase). After key generation, the node keeps its private key and announces the public key in a neighbor advertisement message in response to a neighbor solicitations message and after verification of its neighbors as we will discuss shortly.

B. Key Management Many efforts have been devoted to securing peer communications in wireless ad-hoc networks, and most of them are based on either symmetric-key cryptography (SKC) or public-key cryptography (PKC) systems. Many of them are found to be inadequate for wireless ad-hoc networks, either due to severe communication or computing constraints, or due to the lack of infrastructure support in such networks.

C. Key management is of the greatest interest, since it is a prerequisite for any security procedures of publicly known cryptographic algorithms. For example, in SKC, shared keys or pre-shared secrets should be arranged for involved nodes before they can communicate. In PKC, senders should obtain the public-key of receivers and verify it with trusted

third-parties. For communication in MANETs, nodes need to identify other nodes of their interest. Therefore, mobile nodes can be identified by their own identity of spatial and temporal invariance. For example, nodes propose their identity when joining MANET systems. Nodes should be assisted with additional security procedures to ensure the confidentiality, integrity, and authenticity of their information exchange with intended nodes. Without the help of a trusted key distribution center (KDC), or a trusted certification authority (CA) or any preexisting communication and security infrastructures, nodes may have to deal with unknown relaying nodes without the pre-established trust worthiness, and hence become vulnerable to various passive and active attacks. To overcome this weakness, we base our design on the concept of identity-based key management which serves as a prerequisite for various security procedures. The basic idea is to use an identifier that has a strong cryptographic binding with the public key and components of the mobile node in the same manner that is suggested for MIPv6 in [14]. We will call this identifier, Unique Identifier (UI). This identifier should be owned and used exclusively by the created node. An address (64-bits) that satisfies properties of required UI is obtained as follows:

(a) The most 32-bits refer to the MAC address of the node.

(b) The least 32-bits refer to certain processing on the public key generated by the node at bootstrapping phase, these bits are extracted by:

(1) computing the hash value of the public key using SHA-1,

(2) dividing the hash.value into four parts each of 32-bits, and

(3) performing an XOR operation on the divided hash values and the location of the node, L, used as an evidence. This unique identifier composed of the concatenation of the IP address and the hash value of the public key is secure because an attacker cannot produce a new pair of keys that has the same hash value due to second preimage resistance of one-way hash function, or discover the private key for the given public key. After obtaining the UI, key management mechanism is performed as follows:

(a) The mobile node sends binding update message MSG1 containing the UI described above with a nonce to its corresponding node.

(b) The corresponding node replies with MSG2 containing the same nonce produced by the mobile node.

(c) When receiving MSG2, the mobile node verifies that the nonce is the same as what it was sent in MSG1. It sends MSG3 that contains its public key and the evidence used to generate the UI. This message is signed by the private key of the mobile node.

(d) When the corresponding node receives MSG3, it verifies the signature using the included public key, and verifies that this public key and the evidence produce the same least 32-bits of the UI. Once the message passes the two verifications, it concludes that the mobile node owns this address and the public key. The corresponding node stores the address and the key of the mobile node to be used in further mechanisms.

end-to-end delay.

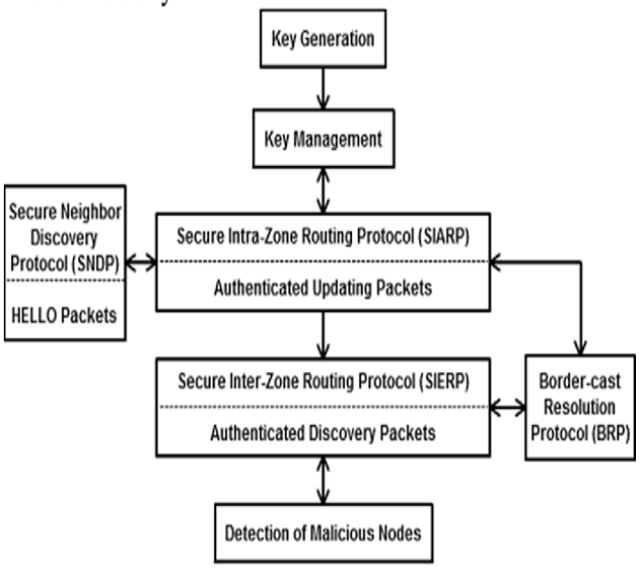

Figure 1. Elements of Secure Zone Routing Protocol (SZRP) [8]

The proposed key management mechanism proposed is efficient since nodes can safely trust the corresponding nodes when they claim ownership of

that identifier. It also will not increase the complexity of the network because:

(1) Not all nodes need to use the mechanisms, only those nodes that wish to perform binding updates,
(2) Not all nodes need to verify MSG3, only those nodes that want to accept the binding update, and
(3) Messages are exchanged directly between the mobile node and its neighbors and are not routed to other nodes. C. Secure Neighbor Discovery In wireless networks, each node needs to know its neighbors to make routing decisions; it stores neighbor information in its routing table that contains the address of the neighbor, and the link state. In MANETs, nodes use neighbor discovery protocol to discover surrounding nodes they can directly communicate with across the wireless channel with signal propagation speed by considering the location or round trip information. Two different nodes, A and B, are considered as neighbors and thus can exchange information directly if and only if the Euclidean distance, |AB|, between them is less than or equal to the neighbor discovery range, R. The NDP protocol relies on HELLO message exchange. Hello messages are used to detect and monitor links to neighbors. If Hello messages are used, each active node periodically broadcasts a Hello message that includes all its neighbors. Because nodes periodically send Hello messages, if a node fails to receive several Hello messages from a neighbor, a link break is detected [3]. The nodes need a correct view of neighbor information which raises the importance of applying a secure neighbor detection protocol. NDP protocol is widely used; however, it can be easily attacked due to lack of security. A malicious node can easily relay or replay packets deluding other nodes that are communicated directly. Many methods have been proposed to protect neighbor information in hostile environments [13]. However, these methods can only protect neighbor relation between benign nodes while compromised nodes can easily circumvent them and setup false relations. In our model, we use a combination of two techniques that rely on time and location based on secure neighbor discovery mechanisms. We based our design on NDP protocol and use the same HELLO message to decrease the number of message flows, and hence the loss of power. Time based protocol (T-based), requires nodes to transmit authenticated messages containing a time-stamp set at the time of sending. Upon receipt of such a message, a receiver checks its freshness by verifying that the message timestamp is within a threshold of the receiver's current time. If so, it accepts the message creator as a neighbor. T-based protocols are not efficient in all cases. For example, they lead to impossibly results if the adversary node has the ability to relay a packet under the predefined threshold value. In time and location based protocols (TL-based), a node requires sending authenticated messages containing a time-stamp set at the time of sending, and their own location. Upon receipt of such a message sent from a node B, the receiver A calculates two estimates; the first estimate is based on the difference of its own clock at reception time and the message's time-stamp. The second one is calculated with the help of the location. If the two distance estimates are equal, A accepts B as a neighbor. The proposed secure NDP protocol consists of three rounds; in the first round the node broadcasts a HELLO message with its location, the time of sending, and the authentication part which indicates that the location and time of sending are authenticated by node A. Authentication process is performed using digital signature with the private key of node A. When the packet is received in the second round, the receiver computes the distance using the location values stored in the packet and transmission time, then, it compares the results obtained with the range of transmission. If the two distance estimates are equal, it verifies the signature. Once the signature is verified, B accepts A as neighbor, signs the packet and replies with beacon acknowledge. Once node A receives the beacon acknowledge, it compares the evidence with the transmitted one; if the two values are equal, it

verifies the signature of the received packet using B's public key. If verification process is checked correctly, node A accepts B as a neighbor, and updates its entire table by assigning a zero value to the trust level of node B. Here, we assumed that corresponding nodes have accurate time and location information based on synchronized clocks and GPS. Inaccurate time and location information can be easily handled by taking into value into four parts each of 32-bits, and (3) performing an XOR operation on the divided hash values and the location of the node, L, used as an evidence. This unique identifier composed of the concatenation of the IP address and the hash value of the public key is secure because an attacker cannot produce a new pair of keys that has the same hash value due to second preimage resistance of one-way hash function, or discover the private key for the given public key. After obtaining the UI, key management mechanism is performed as follows:

(a) The mobile node sends binding update message MSG1 containing the UI described above with a nonce to its corresponding node.

(b) The corresponding node replies with MSG2 containing the same nonce produced by the mobile node.

(c) When receiving MSG2, the mobile node verifies that the nonce is the same as what it was sent in MSG1. It sends MSG3 that contains its public key and the evidence used to generate the UI. This message is signed by the private key of the mobile node.

(d) When the corresponding node receives MSG3, it verifies the signature using the included public key, and verifies that this public key and the evidence produce the same least 32-bits of the UI. Once the message passes the two verifications, it concludes that the mobile node owns this address and the public key. The corresponding node stores the address and the key of the mobile node to be used in further mechanisms. The proposed key management mechanism proposed is efficient since nodes can safely trust the corresponding nodes when they claim ownership of that identifier. It also will not increase the complexity of the network because:

(1) Not all nodes need to use the mechanisms, only those nodes that wish to perform binding updates, (2) Not all nodes need to verify MSG3, only those nodes that want to accept the binding update, and (3) messages are exchanged directly between the mobile node and its neighbors and are not routed to other nodes.

E Secure Neighbor Discovery In wireless networks, each node needs to know its neighbors to make routing decisions; it stores neighbor information in its routing table that contains the address of the neighbor, and the link state. In MANETs, nodes use neighbor discovery protocol to discover surrounding nodes they can directly communicate with across the wireless channel with signal propagation speed by considering the location or round trip information. Two different nodes, A and B, are considered as neighbors and thus can exchange information directly if and only if the Euclidean distance, |AB|, between them is less than or equal to the neighbor discovery range, R. The NDP protocol relies on HELLO message exchange. Hello messages are used to detect and monitor links to neighbors. If Hello messages are used, each active node periodically broadcasts a Hello message that includes all its neighbors. Because nodes periodically send Hello messages, if a node fails to receive several Hello messages from a neighbor, a link break is detected [3]. The nodes need a correct view of neighbor information which raises the importance of applying a secure neighbor detection protocol. NDP protocol is widely used; however, it can be easily attacked due to lack of security. A malicious node can easily relay or replay packets deluding other nodes that are communicated directly. Many methods have been proposed to protect neighbor information in hostile environments [13]. However, these methods can only protect neighbor relation between benign nodes while compromised nodes can easily circumvent them and setup false relations. In our model, we use

a combination of two techniques that rely on time and location based on secure neighbor discovery mechanisms. We based our design on NDP protocol and use the same HELLO message to decrease the number of message flows, and hence the loss of power. Time based protocol (T-based), requires nodes to transmit authenticated messages containing a time-stamp set at the time of sending. Upon receipt of such a message, a receiver checks its freshness by verifying that the message timestamp is within a threshold of the receiver's current time. If so, it accepts the message creator as a neighbor. T-based protocols are not efficient in all cases. For example, they lead to impossibly results if the adversary node has the ability to relay a packet under the predefined threshold value. In time and location based protocols (TL-based), a node requires sending authenticated messages containing a time-stamp set at the time of sending, and their own location. Upon receipt of such a message sent from a node B, the receiver A calculates two estimates; the first estimate is based on the difference of its own clock at reception time and the message's time-stamp. The second one is calculated with the help of the location. If the two distance estimates are equal, A accepts B as a neighbor. The proposed secure NDP protocol consists of three rounds; in the first round the node broadcasts a HELLO message with its location, the time of sending, and the authentication part which indicates that the location and time of sending are authenticated by node A. Authentication process is performed using digital signature with the private key of node A. When the packet is received in the second round, the receiver computes the distance using the location values stored in the packet and transmission time, then, it compares the results obtained with the range of transmission. If the two distance estimates are equal, it verifies the signature. Once the signature is verified, B accepts A as neighbor, signs the packet and replies with beacon acknowledge. Once node A receives the beacon acknowledge, it compares the evidence with the transmitted one; if the two values are equal, it verifies the signature of the received packet using B's public key. If verification process is checked correctly, node A accepts B as a neighbor, and updates its entire table by assigning a zero value to the trust level of node B. Here, we assumed that corresponding nodes have accurate time and location information based on synchronized clocks and GPS. Inaccurate time and location information can be easily handled by taking into Securing Zone Routing Protocol in Ad-Hoc Networks account an acceptable small difference when comparing the estimated values. D. Secure Routing Packets Once we achieve secure information exchange, we can further secure the underlying routing protocol in wireless ad-hoc networks. Security services in MANETs belong to two kinds of massages: the routing massages and the data messages. Both have a different nature and different security needs. We focus here on securing routing because data messages are point-to-point and can be protected with any point-to-point security system. On the other hand, routing messages are sent to intermediate neighbors, processed, possibly modified, and resent. Moreover, as a result of processing of routing message, a node might modify its routing table. This creates the need for both the end-to-end and the intermediate nodes to be able to authenticate the information contained in the routing messages. If all routing messages in MANETs are encrypted with a symmetric cryptography, it means that every member wants to participate in the network has to know the common key. This is the best solution for military networks or any trusted-members network where every member should know the common key before joining the network. However, this is not a suitable solution for a conventional MANET such as meeting room or campus in which members are not trusted [15]. The best option is to use asymmetric cryptography so that the originator of the route messages signs the message. It would not be needed to encrypt the routing messages because they are not secret. The only requirement is that the nodes will be able to detect forged routing messages. To accomplish this

goal we use both digital signature and one-way hash function to attain message authentication, and message integrity as described in more detail below. Secure Intra Zone Routing Protocol To provide packet authentication and message integrity in IARP, digital signature using RSA is used. The IARP packet format. All shaded fields in the packet will be signed using RSA algorithm using the private key of the sender. The signature is stored in the packet before broadcasting it to its neighbors. This signature will provide the authenticity and integrity of the sender and the packet respectively. Secure IARP Scenario Each node periodically advertises its link state (current set of neighbors and corresponding lists of link metrics) through its routing zone. The scope of link state update is controlled by the Time-To-Live (TTL) value that is initialized with the zone radius minus one. The source node signs the whole packet using its private key, appends the signature to the packet, and broadcast it to its surrounding neighbors. Upon receipt of link state update packet, the receiver starts processing the packet if the sender has a high trusted value. Once this is achieved, the receiver creates a copy of the message using the public key of the source already stored in its neighbors' table, and compares the result with the received massage. If the packet passes the verification process, the routing table is recomputed and the packet's TTL value is decremented. The process is repeated as long as the TTL value is greater than zero. Secure Inter Zone Routing Protocol To secure IERP packets, we make end-to-end authentication using digital signature of the non-mutable fields of the packets, the dashed fields of the packet as illustrated in Fig. 3, and a one-way hash function to achieve the integrity of mutable fields while the packets are transmitted through intermediate nodes. The information generated by applying the hash function and the digital signature is transmitted within the packet that we refer to by signature and digest. We use the terms IERP digital signature, and IERP hashing to identify the two mechanisms that are used to secure IERP packets. More details about the functionality of these mechanisms follow. Figure 3: IERP packet format IERP Digital Signature Digital signature using RSA is used to protect the integrity of the non-mutable fields of the packet using the private key of the initiator. The signature is stored in the packet before border-casting it. In order to decrease the overhead on intermediate nodes, the signing process is carried out by the source of the packet in the route request packet and by the destination for the route replay packet. This may lead to a problem in the verification of the route replay. The problem will appear if the RREP packet is generated by an intermediate node which has the link to the destination. To avoid this problem, we restrict the generation of RREP message to the destination only, while intermediate nodes behave as they did not have the route and forward the RREQ message. Although this may lead to significantly increase in the response time, it will decrease the overhead of the verification process. IERP Hashing SZRP uses hashing to attain the integrity of the packets since authentication of data in routing packets is not sufficient, as an attacker could remove a node from the node list. Hashing is performed on the mutable fields of IERP packets, the digest obtained is appended to the packet, and the packet is border-casted. The digest is used to allow every node that receives the message, either an intermediate node or the final destination node, to verify that these fields and especially the route to the destination have not been altered by adversary nodes. Secure IERP Scenario Every time a node requires a route to a destination but does not have the route stored in its route table, it initiates a RREQ packet with the format sets the Query ID to a new identifier that it has not recently used in initiating a route discovery. Query/route source address and query/route destination address are set to the addresses of the source and destination, respectively. The source then computes the digital signature of the non mutable fields and the hash value of its public key, appends them to the signature and digest fields, and border-casts the packet to its peripheral nodes. When any node receives the packet for which it is

not the target node, it checks its local table from recent requests it has received to determine if it has already seen a request from this same source. If it has, the node discards the packet; otherwise, the node checks the node list to be sure that the last node is already a node in its zone with a high trust level. Then, the received node performs hashing on the packet and compares the result with the digest value to verify the integrity of the packet. Once the packet is accepted, the node modifies the request by appending its own address, A, to the node list and replacing the digest field with H[A, digest], which is the hash value, then the node border-casts the packet. When the destination node receives the route request, it checks the authenticity of the RREQ by verifying the signature using the private key of the source. The integrity of the packet is verified by determining that the digest is equal to: H[nn,H[nn-1,H[nn-2,….H[n1, signature]], where n is the number of nodes in the node, ni is the node address at position i in the list. If the destination verifies that the request is valid, it returns a route reply packet to the sender; this packet has the same format of route request packet except the packet type filed. All fields are set to the corresponding values in the same manner as described in the route request phase. This packet is then returned to the source along the source route obtained by reversing the sequence of node list stored in route request packet. Here, there is no need to perform hashing at an intermediate node because it only unicasts the packet to the next hop as listed in the node list. When the source receives the route replay, it verifies the authenticity and integrity of the packet since no changes are added through transmission. If all the verifications are ok, it accepts the packet, otherwise it rejects it. E. Detecting Malicious Nodes Misbehaving nodes can affect network throughput adversely in worst-case scenarios. Most existing ad-hoc routing protocols do not include any mechanism to identify misbehaving nodes. It is necessary to clearly define misbehaving nodes in order to prevent false positives. It may be possible that a node appears to be misbehaving when it is actually encountering a temporary problem such as overload or low battery. Some work has been done to secure ad-hoc networks by using only misbehavior detection schemes. In this kind of approaches, it is too hard to guarantee the integrity and authentication of the routing messages. Therefore, secure routing protocols should provide the integrity and authenticity to the routing messages before being able to identify misbehaving nodes and isolate them during route discovery or updates operations. In our design, we propose a new technique to deal with malicious nodes, and prevent them from further destroying the network. This technique is based on the available information produced by verification processes performed during transferring routing packets. It requires that each node maintains an additional field, trust level, to its neighbors table; this field is dynamically updated with the trust value of the corresponding node. The trust level is initialized with value 3 to indicate that a node is a trusted one. This level is decremented in three cases:

- The node initiates a HELLO message with wrong evidence or does not pass secure neighbor discovery protocol,
- The packet sent by the corresponding node is dropped due to security verification failures.
- The node provides a list with a non-neighbor node.

In all, cases the value is decremented by one. The node is considered as a malicious node if the trust level value reaches zero. The malicious node is transferred to malicious table, and a new authenticated packet, "Alarm Packet", is generated that contains the packet type, the address of the malicious node, and the signature of both. The packet is transmitted in the same manner as IARP packet as described before. Each node that receives the alarm packet reassigns the trust level of the malicious node stored in the packet to zero after verifying the authenticity. In future, each node does not perform any processing on the received packets until verifying the trust level of the sender.

IV. VALIDATION OF SECURITY FUNCTIONALITIES OF SZRP

A. Security Analysis of Digital Signature Digital signature is based on asymmetric key cryptography (RSA), which involves more computational overhead in signing/verifying operations. Most researches claim that digital signature, in general, is less resilient against DoS attacks since an attacker may feed a victim node with a large number of bogus signatures to exhaust the victim's computational resources for verifying them. However, we took this point into account when we designed our protocol. Each node will not verify a message until it verifies the authentication of the transmitted node. Also, a message from a malicious node will not be verified more than three times. After wrong verifications, malicious node will be stored in the black list, and would not be able to consume the resources of this node or other nodes. Digital signature can be verified by any receiver having the public key of the sender. This makes this type applicable for broadcasting messages. Conversely, symmetric key systems and keyed hash functions can be verified only by the intended receiver, making it unappealing for broadcast message authentication, and only used in unicast authentication. Also, this makes digital signature scalable to large numbers of receivers. Only a total number of n public/private key pairs is required compared with symmetric key cryptography or keyed hash functions that require n×(n-1)/2 keys to be maintained in a network with n nodes where establishing these secret keys between any two nodes is a nontrivial problem. One can easily check that secure protocols that are based on shared key are not scalable to large number of nodes, keeping in mind that the processes of managing and distributing these keys will be more complex. B. Security Analysis of RSA System No devastating attacks on RSA have been discovered. Several attacks have been predicted based on week plaintext or weak parameter selections which are not present in our design; the plaintext is strong enough since it has a length of 512 bits.

- RSA is secure against factorization attacks since none of the available factorization algorithms has the ability to factor a large integer; it has a complexity of 2128 which means it needs 298 seconds on a computer that can perform 1-billion bit operations per second.
- RSA is secure against attacks on the encryption exponent because we have used an encryption exponent, e, of 17 bits that is recommended by NIST Special Publication (SP 800-76-1), 2007 [16] to resist all types of this attack such as broadcast attacks, related message attacks, and short pad attacks [17].
- RSA is secure against attacks on the decryption exponent because we have d of 128 bits which is greater than 1/3n1/4 as recommended. However, if the value of d is leaked in any way, the node must immediately change n , e, and d,c. Security Analysis of Unique Identifier Addresses Hash ID Size Consideration In unique identifier addresses, the lower 32 bits are reserved for the IP address, and 32 bits are usable as a hash value. However, the hash function produces 160-bits before performing XOR operations on it. So for the 160 bits, if an attacker tries to find the input that produces the same target output, he should try 2159 possible input values on average; each input with 512 bits long. According to the size of the hash function, we should not be worried about address duplications, this is because we need a population of 1.2×2 160 nodes on average before any two nodes produce duplicate address (according to birthday paradox). Although this is very unlikely, duplicate address detection protocol will detect it, and the node will choose another IP address. Impersonation attacks to UI are also very expensive operation. An attacker must attempt 2 159 tries to find a public key that has the same hash value. If the attacker can perform one million hashes per second, it will need 234 years. Additionally, an attacker must also generate a valid public and private key which is also very expensive as we will discuss

shortly. Key Size Consideration If an attacker finds a RSA public/private key pair that hashes to the same least 32-bits of UI, it can impersonate the mobile node. This can be achieved by a brute force attack. The attacker tries several public keys as input to the hash function used to generate the UI. The difficulty of this attack depends on the size of the modulus n used in generating this public/private key as discussed in the previous section. This is a difficult task because the attacker must generate valid public/private key pairs before performing the hash function. If an attacker can find the public/private key pair that is used to generate the UI, an attacker can impersonate a mobile node and break the RSA system. Thwarting the Effect of Different Types of Attacks After showing that breaking the security of the proposed mechanisms is not an easy task, we will analyze the reaction of our secure protocol in the presence of different kinds of attacks that threaten the routing protocols. We are listing a set of potential attacks where one or multiple nodes could perform in MANETs. In all of the following schemes, {N1, N2, N3} represent cooperative nodes, and {A1, A2} represent attackers. We use four scenarios that present few examples of different types of attacks: modification, dropping, spoofing, and denial of service. Modification Attacks Lengthen/shorten the route: An attacker, A1, between N1 and N2 can receive the RREQ/IARP packets and add itself or a compromised node to the node list of the route in order to make the route going through longer and thus less attractive. Or an attacker can receive RREQ/IARP packets and remove a node from the node list to make the route going through shorter and thus diverts all traffics through it. In SZRP, an attacker cannot add a node to the node list without being an authenticated neighbor to the receiver node. N3 will detect that A1 is not a neighbor, and hence drops the packet before performing any processing on this packet. In case that the attacker is already a neighbor to N3 and can pass neighbor verification mechanism, which rarely happens, verification of the integrity/authenticity of RREQ/IARP will detect that the compromised node or the attacker have been illegally added to the route, and hence the packet will be dropped. Once the packet is dropped, an alarm packet will be sent to all nodes indicating that A1 is an attacker to prevent it from further injecting false packets. This scenario of alarm packets will be repeated whenever a packet is dropped due to verification failure. Deviating the route by modifying DSN: An attacker can receive RREQ sent by the source N1, replay with a greater destination sequence number to N2 which will discard all subsequent traffic destined for the destination N3. Our proposed protocol prevents this type of attacks by restricting the initiating of the RREP packet to the destination who will sign it using its private key. Once the attacker tries to receive the RREP, modify the DSN, it will be detected through verification process of N2, and thus the packet will be dropped. Dropping Attacks A malicious node can decide to drop some or all the packets it has to forward from N1 to N3. This type of attacks cannot be countered in SZRP. However, it does not have a significant impact in dense networks because the control of packet flooding provides the required robustness, e.g., N2 can receive the same packets from N3, or other surrounding nodes. Spoofing Attacks An attacker receiving a RREQ can mislead N2 by generating RREP with less number of nodes in the list other than any legitimate reply. It also will be received with the least delay because of the close distance between the attacker and N2. This type of attacking is thwarted by the disallowance mechanism that prevents any intermediate node from generating RREP because the sender will discard replies except from the destination. If the attacker tries to generate the reply claiming that he is the destination, the generated packet will be discarded because the attacker does not have the private key of the destination and thus cannot generate a valid signature. Replay Attack An attacker might want to mount a replay attack for packets. Replayed requests will be detected at the destination and replayed replies will be detected at source by using standard

mechanisms of the conventional ZRP based on destination sequence numbers and query ID. Impersonating Attacks Impersonating attacks cannot be launched in our SZRP. An attacker that has not compromised any node (and hence does not possess any cryptographic keys from a node) cannot successfully send any routing messages impersonating any other node, since an uncompromised neighbor node will reject the messages due to the failed neighbor authentication. Denial of Service Attacks Denial of Service (DoS) is a very common attack; it may slow down or totally interrupt the overall network. The attacker can use several strategies to achieve this goal and exhaust node resources such as memory and computation resources as the node has to authenticate packet signatures and the digest, while these mechanisms are computationally intensive operations. Beacon Acknowledge Storm: One of DoS attacks is to send a storm of beacon acknowledge messages to a victim node, allowing the node to perform a number of operations. We prevent this type of attack by inserting evidence to the beacon message. If the node receives a beacon acknowledge with an evidence that is not equal to what has been sent in the beacon message the beacon acknowledge will be rejected before performing verification process. IAPR Storm: A malicious node could try to attack its neighbors by sending a storm of IARP update packets with false data to consume the node's resources in computing the new routes, and updating its neighbor table. This type of DoS attack is prevented by using digital signature and detection mechanisms of malicious nodes. Digital signature will check the authenticity of the node and the integrity of the received packets by comparing the node ID with those nodes stored in its neighbor table and performing digital verification using the stored public key of the sender. If three packets are rejected from any cooperative node, an alarm packet will be broadcasted to add this malicious node to the black list. Any received packet from malicious nodes will be dropped without performing any processing on it. This mechanism prevents malicious nodes from further degrading the performance of the network. RREQ Storm: A malicious node could try to attack a node by sending a storm of RREQ packets to a victim node to consume their resources. This type of DoS attack can be easily prevented by using the trust level value of the malicious node as discussed above in IARP storm, or checking the authenticity of the malicious node by the destination node. In both cases, the packet will be rejected if it is proved that the node is not a legitimate one. In general, a malicious node detection mechanism protects the network against all kinds of denial of service attacks. This mechanism is always performed as a first check in order to decrease the overhead produced by signature verification. Once a malicious node is detected, the verifier will drop the packet before performing any farther processing. E. Thwarting the Effects of Well-Known Attacks Rushing Attack: The attacker forwards packets beyond the normal radio transmission range using its higher gain antenna, or a higher power level in order to suppress any subsequent packet. The proposed protocol defends against rushing attack by using secure neighbor detection that allows both the sender and the receiver to verify that the other party is within the normal direct wireless communication range. Wormhole Attack: One of the most severe attacks on MANETs is wormhole attack. The major cause of this attack is the absence of any neighbor detection mechanism. In the wormhole attack, an attacker receives packets at one point in the network, tunnels them to another point in the network, and then replays them into Securing Zone Routing Protocol in Ad-Hoc Networks .the network from that point. The wormhole attack can be detected by an unalterable and independent physical metric, such as time delay or geographical location where both are provided through secure neighbor discovery mechanism. We detect the wormhole attacks through this phase to reduce the overhead and delay produced if the detecting of the wormhole attacks is performed during packet transmission.

V. EXPERIMENTATION AND RESULTS

A. Simulation Environment To evaluate our SZRP in a non-adversarial environment, we have used the Network Simulator 2 (NS-2) [18]. NS-2 is a discrete event simulator written in C++ and OTcl. It was developed by the University of California at Berkeley for simulating the behavior of network and transport layer protocols in a complex network topology. It has been used extensively in evaluating the performance of ad-hoc routing protocols. It realistically models arbitrary node mobility as well as physical radio propagation effects such as signal strength, interference, capture effect, and wireless propagation delay. At the link layer, the simulator implements the complete IEEE 802.11 standard Medium Access Control (MAC) protocol. We modeled our SZRP by modifying the existing ZRP in several ways:

- We increased the packet size to reflect the additional fields necessary to perform security mechanisms. The extended fields hold the public key, the digest, the unique identifier, and the signature. One should note that not all packets hold these fields
- We increased the size of the neighbor table of each node by two fields; the first filed is used to store the public key of its neighbors in each entry, while the other is used to indicate the trust level factor of that neighbor.

- We created a new packet called "Alarm Packet" that is generated and broadcasted to declare malicious nodes when the trusted level value reaches zero. Mobility Model Each node in our experiments moves according to the random waypoint model [19], in which each node begins at a random location and moves independently during the simulation. Each node remains stationary for a specified period that we call the pause time and then moves in a straight line to some new randomly chosen location with a velocity uniformly chosen between 0 and vmax. Once reaching that new location, the node again remains stationary for the pause time, and then chooses a new random location to proceed to at some new randomly chosen velocity, the node continues to repeat this behavior throughout the simulation run. This model can produce large amounts of relative node movements and network topology change, and thus provides a good movement model with which to stress any MANETs routing protocols. This mobility scenario was generated using CMU's TCP/CBR traffic scenario generator. Communication Patterns The data communication pattern in our experiments uses four source-destination pairs, each sending a Constant Bit Rate (CBR) flow of four data packets per second. A rectangular space of size 1500.5 m 2 is used to increase the average number of hops in route used. A rectangular space is recommended in most proposed work to evaluate MANETs' routing protocols as in [6, 7] relative to square space of equal area. It creates a more challenging environment for the routing protocol. Other simulation parameters used are presented in Table I, where we tried to select them similar to other simulations related to secure MANETs protocols [6-10,].

Parameters for studying the performance of SZRP .Performance Metrics We evaluate our proposed protocol by comparing it with the current version of ZRP [2]. Both protocols are run on identical movements and communication scenarios; the primary metrics used for evaluating the performance of SZRP are packet delivery ratio, routing overhead in bytes, routing overhead in packets, and endto-end latency. These metrics are obtained from enhancing the trace files.

- Packet delivery ratio: This is the fraction of the data packets generated by the CBR sources to those delivered to the destination. This evaluates the ability of the protocol to discover routes.
- Routing overhead (bytes): This is the ratio of overhead bytes to the delivered data bytes. The transmission at each hop along the route is counted as one transmission in the calculation of this metric. The routing overhead of a simulation run is calculated as the number of routing bytes generated by the routing agent of all the nodes in the

simulation run. This metric has a high value in secure protocols due to the hash value or signature stored in the packet.

- Routing overhead (packets): This is the ratio of control packet overhead to data packet overhead over all hops. It differs from the routing overhead in bytes since in MANETs if the messages are too large, they will be split into several packets. This metric is always high even in unsecure routing protocols due to control packets used to discover or maintain routes such as IARP and IERP packets.
- Average End-to-End latency: This is the average delay between the sending of data packet by the CBR source and its receipt at the corresponding CBR receiver. This includes all the delays caused during route. acquisition, buffering and processing at intermediate nodes. E. Simulation Results We simulated our SZRP over four scenarios to evaluate it through different movement patterns, network size, transmission rate, and radius of the zone. Performance against Different Mobility Networks In this scenario, we compare the SZRP and ZRP over different values of the pause time. The pause time was changed from 100 s to 500 s to simulate high and low mobility networks. Concerning the packet delivery ratio as a function of pause time, the result shows that the packet delivery ratio obtained using SZRP is above 90% in all scenarios and almost similar to the performance of ZRP. This indicates that the SZRP is highly effective in discovering and maintaining routes for the delivery of data packets, even with relatively high mobility network (low pause time). A network with high mobility nodes has a lower packet delivery ratio because nodes change their location through transmitting data packets that have the predetermined path. For this reason, a high mobility network has a high number of dropped packets due to TTL expiration or link break. For the extra routing overhead introduced by both SZRP and ZRP, where the routing overhead is measured in bytes for both protocols, the results show that the routing overhead of SZRP is significantly higher and increased to nearly 42% for a high mobility network and 27% for a low mobility network. This is due to the increase in size of each packet from the addition of the digest and the signature stored in the packets to verify the integrity and authentication. This routing overhead decreases as the mobility decreases due to increase of the number of updating packets required to keep track of the changes in the topology in order to maintain routing table up-to-date. These packets include both IARP and IERP packets as well as the error messages. We tested the ratio of routing overhead due to control packets transmitted by both protocols in the same simulation environment. The result obtained confirms the previous result of byte overhead. The routing packet decreases for both protocols in the same manner. The ratio of SZRP is higher because of the new messages used in secure neighbor detection schemes as well as the packets produced by splitting the control packets whenever the number of bytes in a packet exceeds a threshold value. Concerning the end-to-end latency for both protocols, the average latency of SZRP is approximately double that of ZRP due to decrease of the available network capacity that is caused by the extra packets and bytes generated for security issues in SZRP. Furthermore, each node has to verify the digital signature and the digest produced by its previous node, compute the newest ones, and insert those values in the packet before retransmission. These signature and hashing processes cause an additional delay in processing the received data packets. The rise in latency at low pause times is due to the non-uniform distribution of nodes in space caused by node motion in the random waypoint. Performance against Different Data Rates and Mobility Patterns In this scenario, we compare the SZRP and ZRP over different values of data rate. We considered these values since high data rate is always an imperative need in any network although it has an extreme effect in increasing the congestion in MANETs. The data rate was changed from one to nine packets per second. These scenarios are performed under high and low mobility networks, 100 s and 500 s, respectively. Fig. 4 shows the packet

delivery ratio of SZRP and ZRP for both low and high mobility networks. We note that the packet delivery ratio exceeds 89% in all cases which can be considered as a good indicator that SZRP goes in the same manner as the conventional ZRP. The delivery packet ratio of low mobility networks increases as the data rate increases as expected since the discovered route to the destination will not change during transmitting the packets, and thus the success of delivering the packet to the same destination will increase. On the other hand, the packet delivery ratio decreases in high mobility networks as the data rate increases because of the high probability of congestion by both the increased data packets and the increased control messages needed to maintain the network nodes up-todate with the changeable topology. Figure 4: Performance of packet delivery ratio against data rate In Fig. 5, the results show that the routing overhead in bytes decreases as the data rate increases. This decrease is related to the increase of data rate all over the time and is not significantly affected by the number of bytes used in securing the control messages. An interesting point that appears in these results is that the number of overhead bytes produced by SZRP is not affected by increasing the data rate which means that the proposed protocol can be applied to network with high and low data rate. Fig. 6 confirms the result obtained in the previous figure; no significant changes are observed since the topology of the network is not changing along different data rates. The routing overhead decreases in both protocols where SZRP in high and low mobility networks still has a higher routing overhead in packets than the conventional ZRP because of the additional packets and bytes used for security purpose. Performance of routing overhead in packets against data rate Performance of average latency against data rate The average end-to-end latency is illustrated in. Both protocols have a lower end-to-end latency in low mobility network. In general, the average latency is constant over the same scenario for low data rate, but it decreases in the high data rate according to the congestion occurred in the network because of the extra data packets sent every second. SZRP with high mobility is worse since it has a higher overhead in routing packets which will cause an earlier congestion. This means that one should be aware when using SZRP in both high mobility and high data rate networks. Performance against Different Network Sizes and Mobility Patterns The third scenario studies the performance of SZRP and ZRP over different network sizes. The number of nodes changes from ten to forty in order to validate our secure routing protocol in different networks. The experiments are performed under high and low mobility rates with data rate of five packets per second. To be consistent, the dimension of the topology used is changed with the same ratio as the number of mobile nodes. Fig. 8 shows the performance of SZRP and ZRP in terms of packet delivery ratio. The SZRP still performs well in low mobility network where it exceeds 99%. However, its performance degrades in a high mobility network. In both cases, the result obtained is accepted because it degrades in the same manner as the conventional ZRP. A final point observed from this figure is that the packet delivery ratio decreases in a large network which is an expected result due to the increase of the traveling time that may lead to TTL expiration. Performance of packet delivery ratio against network size The routing overhead in bytes is shown in for a changeable network size. The results show an increase in total bytes as the network size increases because of the increasing in the number of nodes which leads to escalate the degree of the routing activities in the network; more routing information is shared among the nodes as a result. SZRP has a higher overhead due to the increase of routing packets used to discover or maintain the routes, and the increase of the data needed to perform neighbor discovery mechanisms .Performance of routing overhead in bytes against network size The overall routing overhead in packets .The measurements show that both protocols have an increase in the packets overhead. This is because more nodes are in a

position to generate IARP updating messages and respond to the RREQ messages. However, the increase of SZRP over ZRP in the packets is not too large as in bytes because the extra bytes generated are covered in the existing packets and no extra packets are needed to be generated. The average end-to-end latency .The average latency increases with the increase of network size as well as the dimensions of the topology. SZRP has a higher latency due to processing delay used to provide security requirements. The results provide a clear indication that SZRP will match the performance of ZRP in large networks, because the difference of latency between them decreases as the network size increases in both high and low mobility networks. be at the expense of bytes and packets overhead, it will be acceptable in high bandwidth networks where the high transmission is an essential requirement. Performance of routing overhead in packets against network size. Performance of average latency against network size Performance against Different Routing Zones and Mobility Patterns The last scenario studies the performance of both protocols under different routing zones. The number of routing zone nodes can be regulated through adjustments in each node's transmitter power. To provide adequate network reachability, it is important that a node is connected to a sufficient number of neighbors. However, more is not necessarily better. As the transmitters' coverage areas grow larger, so do the membership of the routing zones, an excessive amount of update traffic may result. SZRP performs well in a different zone radius. It is obvious that both protocols are not affected by the zone radius and still have the ability to discover the route to destination. In low zone radius, the two protocols behave like purely reactive protocol. They depend on route discovery mechanism to find the optimum route to the destination. The overhead produced as for ZRP. It is obvious that the overhead of packets decreases as the zone radius increases until reaching $\rho = 3$ which we can consider here as the optimal radius. Before reaching this value, the protocol behaves around purely reactive protocols where the IERP packets have the majority over all packets. We note that the packets overhead decreases with the increase of zone radius because of border-casting and query control mechanisms that allow queries to be directed to the edge of a routing zone, and thus reducing unnecessary queries within a routing zone. In addition, the packet overhead begins increasing when the zone radius exceeds the optimal because of the route update processes needed to notify neighbors about network topology. In all cases, the routing overhead is increased for high mobility networks because of the extra control packets needed to maintain the changeable locations of nodes. Performance of packet delivery ratio against zone radius Figure 13: Performance of routing overhead in packets against zone radius Figure 14: Performance of routing overhead in bytes against zone radius The results obtained in Fig. 14 confirm the previous discussion, the total overhead in bytes decreases until reaching the optimal zone radius, then it increases again. Both protocols have a higher overhead in a high mobility network because of the extra messages needed to maintain the changes of the topology. The SZRP provides extra overhead in bytes due to extra bytes used in both IARP and IERP packets for security. The difference between the two protocols is smaller in low zone radius. This is because both protocols behave like reactive protocol and the majority of overhead is related to the number of IERP packets which is relatively small since it is generated upon request. So, the extra bytes needed are not too large. Furthermore, the protocols depend on IARP packets in high zone radius which are generated periodically, and need a high number of bytes to provide the security requirements. The average end-to-end latency measured .The purely proactive protocols have the lowest latency because they keep the routing information up-todate at the expense of large portion of the bandwidth. However, low zone radius networks have a higher delay because the nodes need more setup delay to discover the route, SZRP needs more time for extra processing

needed to signing/verifying packets. Performance of average latency against zone radius Effect of Malicious Nodes Behavior The experiments described before compare the performance of SZRP and ZRP when all the nodes in the network are well-behaved. In order to validate our protocol against malicious nodes, we conducted additional experiments to determine the effect of malicious nodes behavior that generate invalid signature caused by any type of attacks discussed earlier. We varied the number of malicious nodes from 0 to 5 nodes. Fig. 16 shows the packet delivery ratio in the presence of malicious nodes. It is obvious that the number of malicious nodes has a significant effect on the packets that are successfully delivered to the destination. The packet delivery ratio is decreased as the number of malicious nodes increases. This is due to the decrease in the available number of nodes that have the ability to provide the route to the destination or establish an alternative one. In general, SZRP still has the ability to deliver packets although the ratio of the malicious nodes reaches 20% of the network size. Effect of malicious modes behaviour

VI. CONCLUSION

This paper is dedicated to demonstrate the security of zone routing protocol; a hybrid protocol that aims to address the problems of excess bandwidth and long route request delay of proactive and reactive routing protocols, respectively. For this purpose, we carefully analyzed the secured protocol proposed with respect to reactive and proactive routing protocols. Four mechanisms are proposed in order to provide a comprehensive secure routing that can defend against all vulnerabilities in ad-hoc networks. The first mechanism is the identity-based key management that does not depend on any trusted key distribution center or certification authority that is rarely found in MANETs. This mechanism provides an identifier that has a strong cryptography binding with the public key of the node. The second mechanism provides a secure neighbor discovery to assure the correct view of neighbor information. It uses a combination of time and location to verify the discovery of legal nodes and prevent a malicious node from deluding other nodes that are within its radio transmission range, and thus preventing most famous attacks such as wormhole, rushing, and replays attacks. The core of the proposed protocol is relying on securing the control packets generated to perform route discovery, route maintenance, and routing tables' updates that provide through the third mechanism to secure routing packets. Both digital signature and one-way hash function are used to achieve our goals. The final mechanism is based on detecting a malicious node using trust level value, followed by using alarm messages to prevent them from further degrading the network performance. Our findings are based on the simulation of SZRP to evaluate its performance with respect to the conventional ZRP using NS simulator under distinguishable scenarios. The selection of parameters and assumptions for each scenario helps in finding the optimal environment. It shows that SZRP has a minimal adverse impact on packet delay and total routing overhead, while the packet delivery ratio achieved is comparable to that of ZRP. Thus, our solution is predicted to become applicable for most systems while the lack of slow execution would not be an issue because of the rapid development of processors. The security analyses presented in this paper emphasize the effectiveness of our secured protocol to provide the required level of security by fulfillment of all security services required by ad-hoc applications such as authentication, integrity, and non-repudiation, and preventing all kinds of attacks threatening ad-hoc networks. Several ideas for future work naturally came up. An enhanced version of SZRP with minor verification will be studied to avoid new attacks that may be performed against this version of SZRP. In addition, a study of the effect of alternative digital signature mechanisms such as elliptic curve can be carried out to reduce the processing time required to perform signing and verification processes. Finally, an environment with

the presence of attackers will be simulated using NS-2 simulator to study the behavior of the current protocols and the enhanced one against all possible attacks.

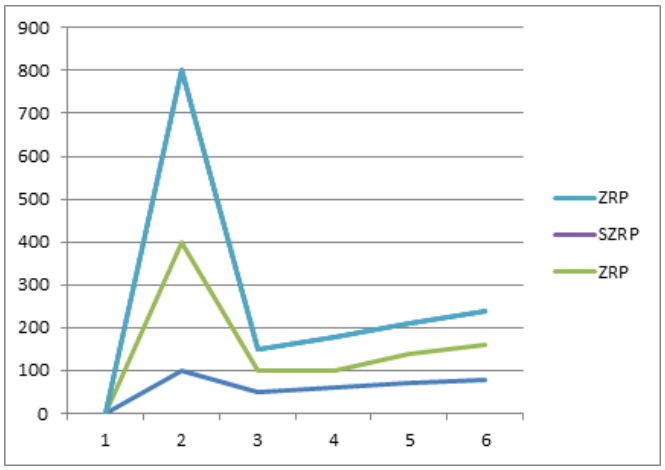

Figure 3. Expermental graph on SZRP

VII. REFERENCES

[1]. A. M. Kamal, "Adaptive Secure Routing in Ad Hoc Mobile Network," M.S. Thesis, Dept. Computer and Systems Science, Royal Institute of Technology, Stockholm, Sweden, 2004.

[2]. Z. J. Haas, M. R. Pearlman, P. Samer, "The Zone Routing Protocol (ZRP) for Ad Hoc Networks," Internet Draft, 2003, available at: http://tools.ietf.org/id/draft-ietf-MANETs-zone-zrp04.txt.

[3]. M. Poturalski, P. Papadimitratos, J. Hubaux, "Secure Neighbor Discovery in Wireless Networks: Formal Investigation of Possibility," in Proc. ACM Symposium on Information, Computer & Communication Security ASIACCS'08, Tokyo, Japan, 2008.

[4]. M Poturalski, P. Papadimitratos, J. Hubaux, "Secure Neighbor Discovery in Wireless Networks," In Proceedings of the 2008 ACM symposium on Information, computer and communications security, Tokyo, Japan, 2008.

[5]. R. Pickholtz, D. Schilling, L.B. Milstein, "Theory of spread spectrum communications – – a tutorial," IEEE Transactions on Communications, v.5, no. 30, pp. 855–884, 1982.

[6]. Y.-C. Hu, D.B. Johnson, A. Perrig, "SEAD: Secure Efficient Distance Vector Routing in Mobile Wireless Ad Hoc Networks," Ad Hoc Networks, 2003, v. 1, pp. 175–192.

[7]. Hu, Yih-Chun, Adrian Perrig, Dave Johnson, "Rushing Attacks and Defense in Wireless Ad Hoc Network Routing Protocols," In Proc. ACM Workshop on Wireless Security, San Diego, WiSe, California, September 2003.

[8]. M. G. Zapata, N. Asokan, "Securing Ad Hoc Routing Protocols," in Proc. ACM Workshop on Wireless Security, Grand Hyatt, WiSe, Singapore, ACM Press, 2002, pp. 1–10.

[9]. P. Papadimitratos, Z. J. Haas, "Secure Link State Routing for Mobile Ad Hoc Networks," in Proc. IEEE Workshop on Security and Assurance in Ad Hoc Networks, IEEE Press, 2003, pp. 27–31.

[10]. K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, E. M. Belding- Royer, "A Secure Routing Protocol for Ad hoc Networks," in Proc. 10th Ann. Int'l Conf. Network Protocols, Paris, ICNP, France, Nov., 2002, pp. 78-87.

[11]. S. Cheung and K. Levitt, "Protecting routing infrastructures from denial of service using cooperative intrusion detection," In Proceedings of the 1997 New Security Paradigms Workshop (September 1998) pp. 94–106.

[12]. Y. -C. Hu, A. Perrig, D. Johnson, "Efficient Security Mechanisms for Routing Protocols," in Proc. Network and Distributed System Security Symp., California, NDPSS, Feb. 2003, pp. 57-73.

[13]. Y. -C. Hu, A. Perrig, D.B. Johnson, "Packet Leashes: A Defense against Wormhole Attacks in Wireless Ad Hoc Networks," in Proc. 22nd Ann. Joint Conf. IEEE Computer and communications Societie

[14]. M. Omar, Y. Challal, and A. Bouabdallah, "Reliable and fully distributed trust model for

mobile ad hoc networks," Computers & Security, vol. 28, pp. 199 – 214, 2009.

[15]. A. Adnane, C. Bidan, and R. T. de Sousa Junior, "Trustbased security for the olsr routing protocol," Computer Communications, vol. 36, no. 10, pp. 1159–1171, 2013

[16]. M. Marimuthu and I. Krishnamurthi, "Enhanced olsr for defense against dos attack in ad hoc networks," Communications and Networks, Journal of, vol. 15, no. 1, pp. 31–37, Feb 2013.

[17]. Y. Jeon, T.-H. Kim, Y. Kim, and J. Kim, "Lt-olsr: Attacktolerant olsr against link spoofing," in Proceedings of the 2012 IEEE 37th Conference on Local Computer Networks (LCN 2012), ser. LCN '12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 216–219.

[18]. D. Malik, K. Mahajan, and M. Rizvi, "Security for node isolation attack on olsr by modifying mpr selection process," in Networks Soft Computing (ICNSC), 2014 First International Conference on, Aug 2014, pp. 102–106.

[19]. A. Adnane, C. Bidan, and R. de Sousa, "Trust-based countermeasures for securing olsr protocol," in Computational Science and Engineering, 2009. CSE '09. International Conference on, vol. 2, Aug 2009, pp. 745– 752

[20]. P. Suresh, R. Kaur, M. Gaur, and V. Laxmi, "Collusion attack resistance through forced mpr switching in olsr," in Wireless Days (WD), 2010 IFIP, Oct 2010, pp. 1–5.

[21]. G. Zapata, N. Asokan, "Securing Ad Hoc Routing Protocols," in Proc. ACM Workshop on Wireless Security, Grand Hyatt, WiSe, Singapore, ACM Press, 2002, pp. 1–10.

[22]. P. Papadimitratos, Z. J. Haas, "Secure Link State Routing for Mobile Ad Hoc Networks," in Proc. IEEE Workshop on Security and Assurance in Ad Hoc Networks, IEEE Press, 2003, pp. 27–31.

[23]. K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, E. M. Belding- Royer, "A Secure Routing Protocol for Ad hoc Networks," in Proc. 10th Ann. Int'l Conf. Network Protocols, Paris, ICNP, France, Nov., 2002, pp. 78-87.