International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

EAACK – To Overcome from Intruders Attacks in Manet's by Providing Security Checks

H. Syed Siddiq[1], M. Hymavathi[2]

[1]M.Tech Student, Department of Computer Science and Engineering, Quba College of Engineering and Technology, Venkatachalam, Nellore District, Andhra Pradesh, India

[2]Associate Professor, Department of Computer Science and Engineering, Quba College of Engineering and Technology, Venkatachalam, Nellore District, Andhra Pradesh, India

Abstract: ***Mobile Ad hoc NETwork (MANET) is one of the most important and unique applications. MANET does not require a fixed network infrastructure. Every single node works as both a transmitter and a receiver. Nodes communicate directly with each other when they are both within the same communication range. Otherwise, they rely on their neighbors to relay messages. The open medium and wide distribution of nodes make MANET vulnerable to malicious attackers. A new intrusion-detection system named Enhanced Adaptive ACKnowledgment (EAACK) specially designed for MANETs. EAACK demonstrates higher malicious-behavior-detection rates in certain circumstances while does not greatly affect the network performances.***

Keywords: Mobile Ad hoc NETwork (MANET), Enhanced Adaptive ACKnowledgement (EAACK), Packet Delivery Ratio (PDR), Digital Signature Algorithm (DSA), Secure ACKnowledgement (S-ACK).

1. Introduction

A mobile ad hoc network (MANET) is a self-configuring less infrastructure network of mobile devices connected by wireless. Ad hoc is Latin and means "for this purpose". Each device in a MANET is free to move independently in any direction, and will therefore change its links to other devices frequently. Each must forward traffic unrelated to its own use, and therefore be a router. The primary challenge in building a MANET is equipping each device to continuously maintain the information required to properly route traffic. Such networks may operate by themselves or may be connected to the larger Internet. A Mobile Ad hoc NETwork (MANET) is a system of wireless mobile nodes that dynamically self-organize in arbitrary and temporary network topologies. In the mobile ad hoc network, nodes can directly communicate with all the other nodes within their radio ranges; whereas nodes that not in the direct communication range use intermediate node(s) to communicate with each other. In these two situations, all the nodes that have participated in the communication automatically form a wireless network, therefore this kind of wireless network can be viewed as mobile ad hoc network. The mobile ad hoc network has the following typical features:

a) Unreliability of wireless links between nodes. Because of the limited energy supply for the wireless nodes and the mobility of the nodes, the wireless links between mobile nodes in the ad hoc network are not consistent for the communication participants.
b) Constantly changing topology. Due to the continuous motion of nodes, the topology of the mobile ad hoc network changes constantly.
c) Lack of incorporation of security features in statically configured wireless routing protocol not meant for ad hoc environments. Because the topology of the ad hoc networks is changing constantly.

A MANETS are mobile; they use wireless connections to connect to various networks. This can be a standard Wi-Fi connection, or another medium, such as a cellular or satellite transmission.

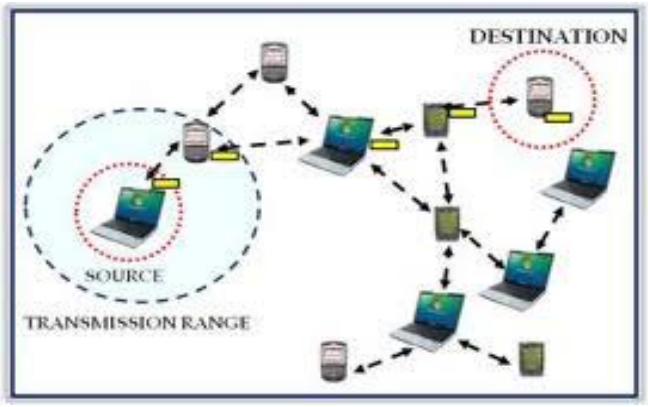


Figure 1: Structure of MANET

2. Related Work and Background

2.1 Intrusion Detection System in MANETs

Due to the limitations of most MANET routing protocols, nodes in MANETs assume that other nodes always cooperate with each other to relay data. This assumption leaves the attackers with the opportunities to achieve significant impact on the network with just one or two compromised nodes. To address this problem, IDS should be added to enhance the security level of MANETs. If MANET can detect the attackers as soon as they enter the network, we will be able to completely eliminate the potential damages caused by compromised nodes at the first time. In this section, we mainly describe three existing approaches, namely, Watchdog, TWOACK, and Adaptive ACKnowledgment (AACK).

2.1.1 Watchdog

Watchdog improves the throughput of network with the presence of malicious nodes. It is responsible for detecting

malicious node misbehaviors in the network. Watchdog detects malicious misbehaviors by promiscuously listening to its next hop's transmission. If a Watchdog node overhears that its next node fails to forward the packet within a certain period of time, it increases its failure counter. Whenever a node's failure counter exceeds a predefined threshold, the Watchdog node reports it as misbehaving. In this case, the Path rater cooperates with the routing protocols to avoid the reported nodes in future transmission. Many following research studies and implementations have proved that the Watchdog scheme is efficient. Furthermore, compared to some other schemes, Watchdog is capable of detecting malicious nodes rather than links. These advantages have made the Watchdog scheme a popular choice in the field. The Watchdog scheme fails to detect malicious misbehaviors with the presence of the following: 1) ambiguous collisions; 2) receiver collisions; 3) limited transmission power; 4) false misbehavior report; 5) collusion; and 6) partial dropping.

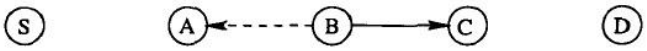


Figure 2: Working mechanism of watchdog

2.1.2 TWOACK:

With respect to the six weaknesses of the Watchdog scheme, many researchers proposed new approaches to solve these issues. TWOACK is one of the most important approaches among them. Aiming to resolve the receiver collision and limited transmission power problems of Watchdog, TWOACK detects misbehaving links by acknowledging every data packet transmitted over every three consecutive nodes along the path from the source to the destination. Upon retrieval of a packet, each node along the route is required to send back an acknowledgment packet to the node that is two hops away from it down the route. TWOACK is required to work on routing protocols such as Dynamic Source Routing (DSR). The working process of TWOACK is Node A first forwards Packet 1 to node B, and then, node B forwards Packet 1 to node C. When node C receives Packet 1, as it is two hops away from node A, node C is obliged to generate a TWOACK packet, which contains reverse route from node A to node C, and sends it back to node A. The retrieval of this TWOACK packet at node A indicates that the transmission of Packet 1 from node A to node C is successful. Otherwise, if this TWOACK packet is not received in a predefined time period, both nodes B and C are reported malicious. The same process applies to every three consecutive nodes along the rest of the route. The TWOACK scheme successfully solves the receiver collision and limited transmission power problems posed by Watchdog. However, the acknowledgment process required in every packet transmission process added a significant amount of unwanted network overhead. Due to the limited battery power nature of MANETs, such redundant transmission process can easily degrade the life span of the entire network.

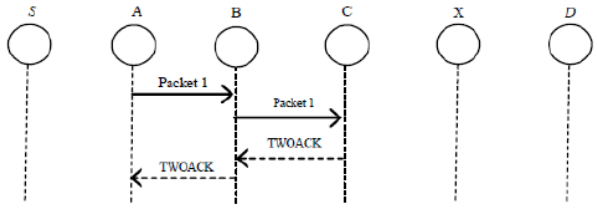


Figure 3: Working mechanism of Two ACK

2.1.3 AACK

Compared to TWOACK, AACK significantly reduced network overhead while still capable of maintaining or even surpassing the same network throughput. In the ACK scheme, the source node S sends out Packet 1 without any overhead except 2 b of flag indicating the packet type. All the intermediate nodes simply forward this packet. When the destination node D receives Packet 1, it is required to send back an ACK acknowledgment packet to the source node S along the reverse order of the same route. Within a predefined time period, if the source node S receives this ACK acknowledgment packet, then the packet transmission from node S to node D is successful. Otherwise, the source node S will switch to TACK scheme by sending out a TACK packet. The concept of adopting a hybrid scheme in AACK greatly reduces the network overhead, but both TWOACK and AACK still suffer from the problem that they fail to detect malicious nodes with the presence of false misbehavior report and forged acknowledgment packets. In fact, many of the existing IDSs in MANETs adopt an acknowledgment-based scheme, including TWOACK and AACK. The functions of such detection schemes all largely depend on the acknowledgment packets. Hence, it is crucial to guarantee that the acknowledgment packets are valid and authentic. To address this concern, we adopt a digital signature in our proposed scheme named Enhanced AACK (EAACK).

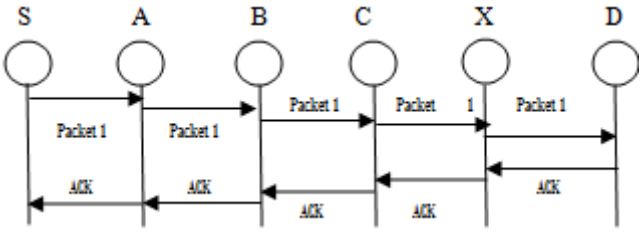


Figure 4: Working mechanism of AACK

2.2 Digital Signature

Digital signatures have always been an integral part of cryptography in history. Digital signature schemes can be mainly divided into the following two categories.

2.2.1 Digital signature with appendix

The original message is required in the signature verification algorithm.

2.2.2 Digital signature with message recovery

This type of scheme does not require any other information besides the signature itself in the verification process.

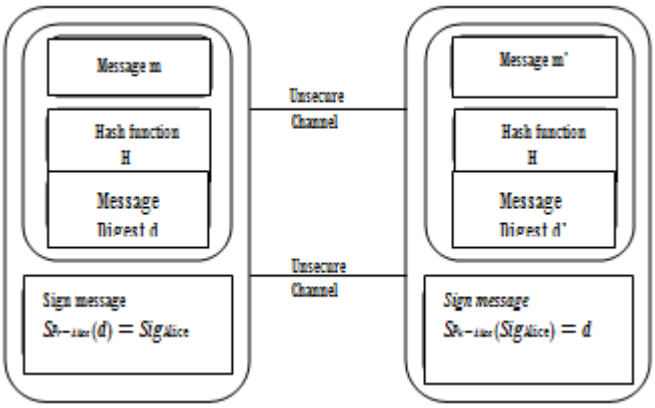


Figure 5: Communication with digital signature.

In this research work, we implemented both DSA and RSA in our proposed EAACK scheme. The main purpose of this implementation is to compare their performances in MANETs. The general flow of data communication with digital signature is shown in Fig 5.

First, a fixed-length message digest is computed through a hash function *H* for every message *m*. This process can be described as $H(m) = d$. (1)

Second, the sender Alice needs to apply its own private key $Pr{-}Alice$ on the computed message digest *d*. The result is a signature *Sig*Alice, which is attached to message *m* and Alice's secret private key $SPr{-}Alice$ (*d*) = *Sig*Alice. (2)

To ensure the validity of the digital signature, the sender Alice is obliged to always keep her private key $Pr{-}Alice$ as a secret without revealing to anyone else. Otherwise, if the attacker Eve gets this secret private key, she can intercept the message and easily forge malicious messages with Alice's signature and send them to Bob. As these malicious messages are digitally signed by Alice, Bob sees them as legit and authentic messages from Alice. Thus, Eve can readily achieve malicious attacks to Bob or even the entire network. Next, Alice can send a message *m* along with the signature *Sig*Alice to Bob via an unsecured channel. Bob then computes the received message *m*_ against the hash function *H* to get the message digest *d*_. This process can be generalized as $H(m\_) = d\_$. (3)

Bob can verify the signature by applying Alice's public key $Pk{-}Alice$ on *Sig*Alice, by using $SPk{-}Alice$ (*Sig*Alice) = *d*. (4)
If *d* == *d*_, then it is safe to claim that the message *m*_ transmitted through an unsecured channel is indeed sent from Alice and the message itself are intact.

3. Problem Definition

Our proposed approach EAACK is designed to tackle three of the six weaknesses of Watchdog scheme, namely, false misbehavior, limited transmission power, and receiver collision. In this section, we discuss these three weaknesses in detail.

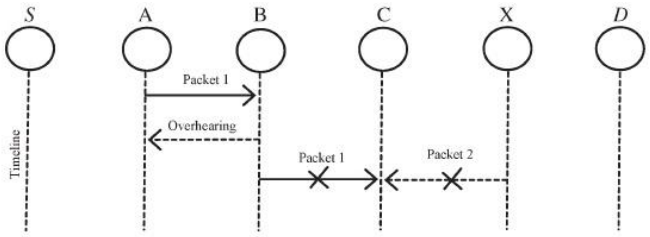


Figure 6: Receiver collisions: Both nodes B and X is trying to send Packet 1 and Packet 2, respectively, to node C at the same time.

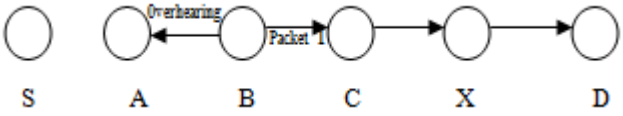


Figure 7: Limited transmission power: Node B limits its transmission power so that the packet transmission can be overheard by node A but too weak to reach node C.

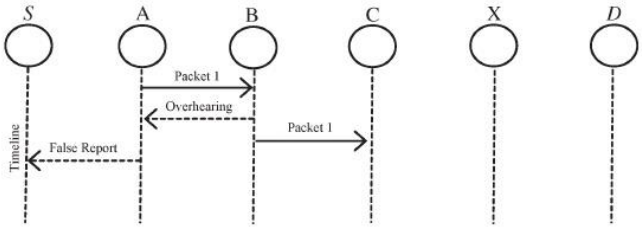


Figure 8: False misbehavior report: Node A sends back a misbehavior report even though node B forwarded the packet to node C.

4. Scheme Description

In this section, we describe our proposed EAACK scheme in detail. In this paper, we extend it with the introduction of digital signature to prevent the attacker from forging acknowledgment packets. EAACK is consisted of three major parts, namely, ACK, secure ACK (S-ACK), and misbehavior report authentication (MRA). In order to distinguish different packet types in different schemes, we included a 2-b packet header in EAACK. According to the Internet draft of DSR [11], there is 6 b reserved in the DSR header. In EAACK, we use 2 b of the 6 b to flag different types of packets.

Please note that, in our proposed scheme, we assume that the link between each node in the network is bidirectional. Furthermore, for each communication process, both the source node and the destination node are not malicious. Unless specified, all acknowledgment packets described in this research are required to be digitally signed by its sender and verified by its receiver.

4.1 ACK

As discussed before, ACK is basically an end-to-end acknowledgment scheme. It acts as a part of the hybrid scheme in EAACK, aiming to reduce network overhead when no network misbehavior is detected. In Fig. 9, in ACK mode, node S first sends out an ACK data packet *Pad*1 to the destination node D. If all the intermediate nodes along the

route between nodes S and D are cooperative and node D successfully receives *Pad*1, node D is required to send back an ACK acknowledgment packet *Pak*1 along the same route but in a reverse order. Within a predefined time period, if node S receives *Pak*1, then the packet transmission from node S to node D is successful. Otherwise, node S will switch to S-ACK mode by sending out an S-ACK data packet to detect the misbehaving nodes in the route.

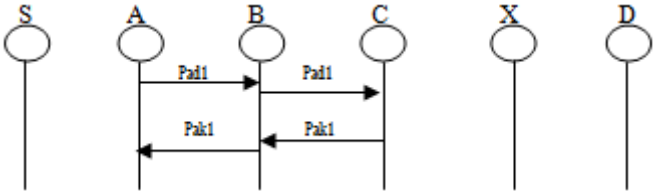


Figure 9: ACK scheme: The destination node is required to send back an acknowledgment packet to the source node when it receives a new packet.

4.2 S-ACK

The S-ACK stands for secure acknowledgement. This scheme is an improved version of the TWOACK scheme. The principle is to let every three consecutive nodes work in a group to detect misbehaving nodes. For every three consecutive nodes in the route, the third node is required to send an S-ACK acknowledgment packet to the first node. The intention of introducing S-ACK mode is to detect misbehaving nodes in the presence of receiver collision or limited transmission power. As shown in Fig. 10, in S-ACK mode, the three consecutive nodes (i.e., A, B, and C) work in a group to detect misbehaving nodes in the network. Node A first sends out S-ACK data packet *Psad*1 to node B. Then, node B forwards this packet to node C. When node C receives *Psad*1, as it is the third node in this three-node group, node C is required to send back an S-ACK acknowledgment packet *Psak*1 to node B. Node B forwards *Psak*1 back to node A. If node A does not receive this acknowledgment packet within a predefined time period, both nodes B and C are reported as malicious. Moreover, a misbehavior report will be generated by node A and sent to the source node S. Nevertheless, unlike the TWOACK scheme, where the source node immediately trusts the misbehavior report, EAACK requires the source node to switch to MRA mode and confirm this misbehavior report. This is a vital step to detect false misbehavior report in our proposed scheme.

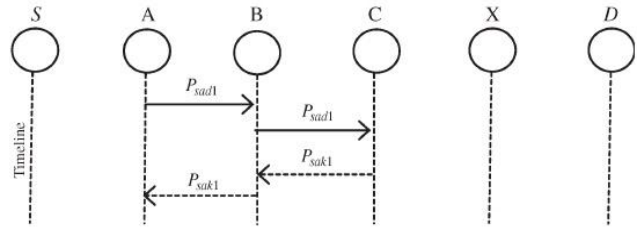


Figure 10: s-ack scheme node C is required to send back an acknowledge packet t o node B

4.3 MRA

MRA stands for Misbehavior report authentication. The MRA scheme is designed to resolve the weakness of Watchdog when it fails to detect misbehaving nodes with the presence of false misbehavior report. The false misbehavior report can be generated by malicious attackers to falsely report innocent nodes as malicious. This attack can be lethal to the entire network when the attackers break down sufficient nodes and thus cause a network division. The core of MRA scheme is to authenticate whether the destination node has received the reported missing packet through a different route. To initiate the MRA mode, the source node first searches its local knowledge base and seeks for an alternative route to the destination node. If there is no other that exists, the source node starts a DSR routing request to find another route. Due to the nature of MANETs, it is common to find out multiple routes between two nodes. By adopting an alternative route to the destination node, we circumvent the misbehavior reporter node. When the destination node receives an MRA packet, it searches its local knowledge base and compares if the reported packet was received. If it is already received, then it is safe to conclude that this is a false misbehavior report and whoever generated this report is marked as malicious. Otherwise, the misbehavior report is trusted and accepted. By the adoption of MRA scheme, EAACK is capable of detecting malicious nodes despite the existence of false misbehavior report.

4.4 Digital Signature

As discussed before, EAACK is an acknowledgment-based IDS. All three parts of EAACK, namely, ACK, S-ACK, and MRA, are acknowledgment-based detection schemes. They all rely on acknowledgment packets to detect misbehaviors in the network. Thus, it is extremely important to ensure that all acknowledgment packets in EAACK are authentic and untainted. Otherwise, if the attackers are smart enough to forge acknowledgment packets, all of the three schemes will be vulnerable.

With regard to this urgent concern, we incorporated digital signature in our proposed scheme. In order to ensure the integrity of the IDS, EAACK requires all acknowledgment packets to be digitally signed before they are sent out and verified until they are accepted. However, we fully understand the extra resources that are required with the introduction of digital signature in MANETs. To address this concern, we implemented both DSA and RSA digital signature schemes in our proposed approach. The goal is to find the most optimal solution for using digital signature in MANETs.

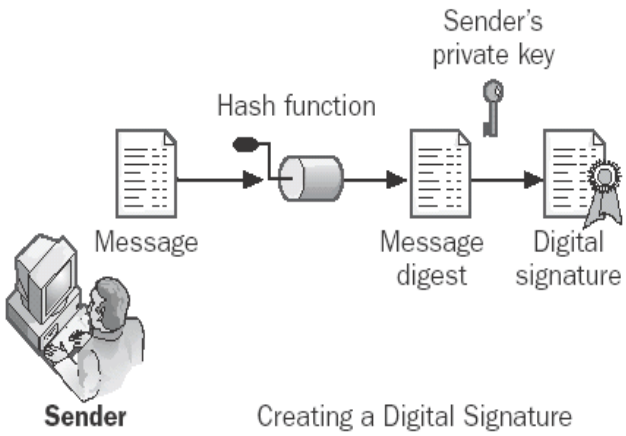


Figure 11: Process in Digital signature

The first part of the DSA algorithm is the public key and private key generation, which can be described as:

- Choose a prime number q, which is called the prime divisor.
- Choose another primer number p, such that p-1 mod q = 0. p is called the prime modulus.
- Choose an integer g, such that 1 < g < p, g**q mod p = 1 and g = h**((p–1)/q) mod p. q is also called g's multiplicative order modulo p.
- Choose an integer, such that 0 < x < q.
- Compute y as g**x mod p.
- Package the public key as {p,q,g,y}.
- Package the private key as {p,q,g,x}.

The second part of the DSA algorithm is the signature generation and signature verification, which can be described as:

To generate a message signature, the sender can follow these steps:

- Generate the message digest h, using a hash algorithm like SHA1.
- Generate a random number k, such that 0 < k < q.
- Compute r as (g**k mod p) mod q. If r = 0, select a different k.
- Compute i, such that k*i mod q = 1. i is called the modular multiplicative inverse of k modulo q.
- Compute s = i*(h+r*x) mod q. If s = 0, select a different k.
- Package the digital signature as {r,s}.

To verify a message signature, the receiver of the message and the digital signature can follow these steps:

- Generate the message digest h, using the same hash algorithm.
- Compute w, such that s*w mod q = 1. w is called the modular multiplicative inverse of s modulo q.
- Compute u1 = h*w mod q.
- Compute u2 = r*w mod q.
- Compute v = (((g**u1)*(y**u2)) mod p) mod q.
- If v == r, the digital signature is valid.

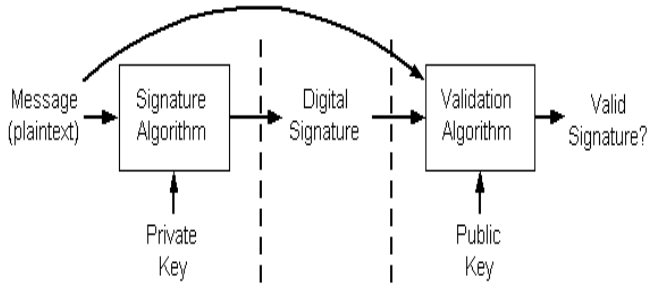


Figure 12: Digital signature

5. Results

Our simulation is conducted within the Network Simulator(NS) 2.34 environment on a platform with GCC 4.3 and Ubuntu 9.10. The system is running on a laptop with Core 2 Duo T7250 CPU and 3-GB RAM. In order to better compare our simulation results with other research works, we adopted the default scenario settings in NS 2.34. The intention is to provide more general results and make it easier for us to compare the results. In NS 2.34, the default configuration specifies 50 nodes in a flat space with a size of 670 × 670 m. The maximum hops allowed in this configuration setting are four. Both the physical layer and the 802.11 MAC layer are included in the wireless extension of NS2. The moving speed of mobile node is limited to 20 m/s and a pause time of 1000 s. User Datagram Protocol traffic with constant bit rate is implemented with a packet size of 512 B. For each scheme, we ran every network scenario three times and calculated the average performance. In order to measure and compare the performances of our proposed scheme, we continue to adopt the following two performance metrics.

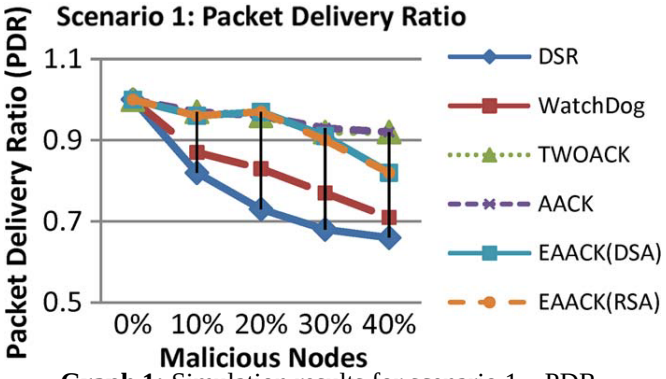
1) Packet delivery ratio (PDR): PDR defines the ratio of the number of packets received by the destination node to the number of packets sent by the source node.

2) Routing overhead (RO): RO defines the ratio of the amount of routing-related transmissions [Route REQuest (RREQ), Route REPly (RREP), Route ERRor (RERR), ACK, S-ACK, and MRA].

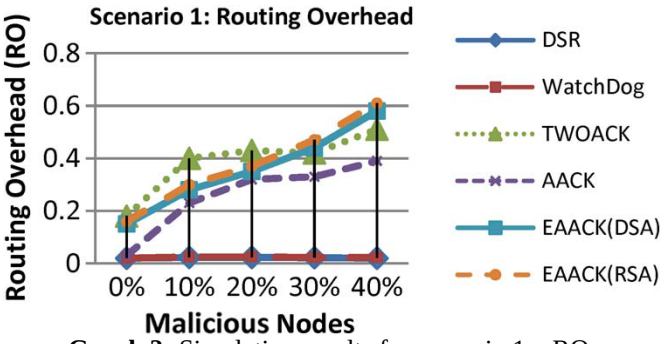
During the simulation, the source route broadcasts an RREQ message to all the neighbors within its communication range. Upon receiving this RREQ message, each neighbor appends their addresses to the message and broadcasts this new message to their neighbors. If any node receives the same RREQ message more than once, it ignores it. Regarding the digital signature schemes, we adopted an open source library named Botan. This cryptography library is locally compiled with GCC 4.3. To compare performances between DSA and RSA schemes, we generated a 1024-b DSA key and a 1024-b RSA key for every node in the network. We assumed that both a public key and a private key are generated for each node and they were all distributed in advance. The typical sizes of public- and private-key files are 654 and 509 B with a 1024-b DSA key, respectively. On the other hand, the sizes of public- and private-key files for 1024-b RSA are 272 and 916 B, respectively. The signature file sizes for DSA and RSA are 89 and 131 B, respectively. One of the most popular sensor nodes in the market is Tmote Sky [34]. This type of sensor is equipped with a TIMSP430F1611 8-MHz CPU and 1070 KB of memory space. We believe that this is enough for handling our simulation settings in terms of both computational power and memory space.

Table 1: Table of experiment

Scenario 1: Packet Delivery Ratio					
	Malicious Nodes: 0%	Malicious Nodes: 10%	Malicious Nodes: 20%	Malicious Nodes: 30%	Malicious Nodes: 40%
DSR	1	0.82	0.73	0.68	0.66
Watchdog	1	0.83	0.77	0.7	0.67
TWOACK	1	0.97	0.96	0.92	0.92
AACK	1	0.96	0.96	0.93	0.92
EAACK(DSA)	1	0.96	0.97	0.93	0.91
EAACK(RSA)	1	0.96	0.97	0.92	0.92
Scenario 1: Routing Overhead					
	Malicious Nodes: 0%	Malicious Nodes: 10%	Malicious Nodes: 20%	Malicious Nodes: 30%	Malicious Nodes: 40%
DSR	0.02	0.023	0.023	0.022	0.02
Watchdog	0.02	0.025	0.025	0.023	0.023
TWOACK	0.18	0.4	0.43	0.42	0.51
AACK	0.03	0.23	0.32	0.33	0.39
EAACK(DSA)	0.15	0.28	0.35	0.44	0.58
EAACK(RSA)	0.16	0.3	0.37	0.47	0.61



Graph 1: Simulation results for scenario 1—PDR.



Graph 2: Simulation results for scenario 1—RO.

6. Conclusion

Packet-dropping attack has always been a major threat to the security in MANETs. In this research paper, we have proposed a novel IDS named EAACK protocol specially designed for MANETs and compared it against other popular mechanisms in different scenarios through simulations. The results demonstrated positive performances against Watchdog, TWOACK, and AACK in the cases of receiver collision, limited transmission power, and false misbehavior report. Furthermore, in an effort to prevent the attackers from initiating forged acknowledgment attacks, we extended our research to incorporate digital signature in our proposed scheme. Digital signature algorithms are used to provide authentication of data and validating the sender. we implemented both DSA and RSA schemes in our simulation. Eventually, we arrived to the conclusion that the DSA scheme is more suitable to be implemented in MANETs.

7. Future Scope

To increase the merits of our research work, we plan to investigate the following issues in our future research:

- Testing the performance of EAACK in real network environment instead of software simulation.
- Possibilities of adopting hybrid cryptography techniques to further reduce the network overhead caused by digital signature.
- Examine the possibilities of adopting a key exchange mechanism to eliminate the requirement of predistributed keys.

Reference

[1] K. Al Agha, M.-H. Bertin, T. Dang, A. Guitton, P. Minet, T. Val, and J.-B. Viollet, "Which wireless technology for industrial wireless sensor networks? The development of OCARI technol," *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4266–4278, Oct. 2009.

[2] R. Akbani, T. Korkmaz, and G. V. S. Raju, "Mobile Ad hoc NetworkSecurity," in *Lecture Notes in Electrical Engineering*, vol. 127. New York: Springer-Verlag, 2012, pp. 659–666.

[3] K. Liu, J. Deng, P. K. Varshney, and K. Balakrishnan, "An acknowledgment-based approach for the detection of routing misbehavior in MANETs," *IEEE Trans. Mobile Comput.*, vol. 6, no. 5, pp. 536–550, May 2007.

[4] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proc. 6th Annu. Int. Conf. Mobile Comput. Netw.*, Boston, MA, 2000, pp. 255–265.

[5] N. Nasser and Y. Chen, "Enhanced intrusion detection systems for discovering malicious nodes in mobile ad hoc network," in *Proc. IEEE Int. Conf. Commun.*, Glasgow, Scotland, Jun. 24–28, 2007, pp. 1154–1159.

[6] J. Parker, J. Undercoffer, J. Pinkston, and A. Joshi, "On intrusion detection and response for mobile ad hoc networks," in *Proc. IEEE Int. Conf. Perform., Comput., Commun.*, 2004, pp. 747–752.

[7] A. Patwardhan, J. Parker, A. Joshi, M. Iorga, and T. Karygiannis, "Secure routing and intrusion detection in ad hoc networks," in *Proc. 3rd Int. Conf. Pervasive Comput. Commun.*, 2005, pp. 191–199.
[8] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1983.
[9] B. Sun, "Intrusion detection in mobile ad hoc networks," Ph.D. dissertation, Texas A&M Univ., College Station, TX, 2004.
[10] K. Stanoevska-Slabeva and M. Heitmann, "Impact of mobile ad-hoc networks on the mobile value system," in *Proc. 2nd Conf. m-Bus.*, Vienna, Austria, Jun. 2003.
[11] M. Zapata and N. Asokan, "Securing *ad hoc* routing protocols," in *Proc. ACM Workshop Wireless Secur.*, 2002, pp. 1–10.
[12] L. Zhou and Z. Haas, "Securing ad-hoc networks," *IEEE Netw.*, vol. 13, no. 6, pp. 24–30, Nov./Dec. 1999.

Author Profile

H Syed Siddiq received the B.Tech degree in Computer Science and Engineering from Quba College of Engineering and Technology; Nellore in 2012. He is currently pursuing his M.Tech in the same stream in Quba College of Engineering and Technology, Nellore.

Hymavathi.M did M.Tech(CSE) in JNTU , Anantapur. And currenty working as a Associate professor in CSE department in Quba College of Engineering and Technology, Venkatachalam Mandal, Nellore, AP, India.