

Threshold Secret Sharing Authentication Scheme in Noisy Mobile Ad Hoc Networks

Shakir M. Hussain¹, Hussein Al-Bahadili², Ghassan Issa¹, and Khalid Al-Zayyat³

¹ Faculty of Information Technology, Petra University, Jordan

²The Arab Academy for Banking & Financial Sciences, Jordan

³ Depaul University, Jordan

Abstract - This paper presents a description and performance evaluation of a threshold secret sharing (TSS) authentication scheme for self-securing mobile ad hoc networks (MANETs) suffering from high packet-loss and node mobility. In order to evaluate the performance of the TSS scheme in a noisy MANET, a number of simulations were carried-out. We concluded that presence of noise inflicts significant reduction in the authentication success ratio (S_R) and consequently degrades the performance of the network, while node mobility inflicts no or insignificant effects. The outcomes of these simulations are so important to facilitate efficient network management.

Keywords: node authentication; certification authority; MANET; threshold secret sharing.

1. Introduction

Security design in mobile ad hoc networks (MANETs) is challenging for several reasons, such as [1]: security breach, mobility and service ubiquity, network dynamics, network scale, etc. MANETs are very vulnerable to a number of security attacks, such as: passive eavesdropping over wireless channel, denial-of-service (DoS) attacks by malicious nodes, and attacks from compromised entities or stolen devices [2]. The main requirements that need to be carefully considered to ensure high-level of MANETs security are: confidentiality, authentication, integrity, availability, and non-repudiation.

This paper is concerned with one of the main security requirements for MANETs, namely authentication. The most popular network authentication architectures are Kerberos [3], the X.509 standard [4], and public-key infrastructure (PKI) trust model [5], which are based on using a globally trusted certificate authority (CA) model [6]. Using a globally trusted CA model may work well in wired or infrastructure wireless networks, but not MANETs because: MANETs provide no infrastructure support, each of the CA servers is exposed to a single point of compromises and failures, multihop communications over the error-prone wireless channel expose data transmissions to high packet-loss rate and large latency, and

frequent route changes induced by node mobility, which makes locating and contacting CA servers in a timely fashion non-trivial [7]. Although, variations of the CA model, such as hierarchical CAs and CA delegations can ameliorate, but cannot address issues such as service availability and robustness [5]. Therefore, more efficient and reliable solutions are required to address the above issues. One alternative solution is to use the concept of threshold secret sharing (TSS) scheme proposed by Adi Shamir in 1978 [8].

A TSS-based authentication is the most suitable scheme for securing wireless ad hoc networks. A number of researches have been carried-out to investigate its performance in terms of authentication success ratio, average delay, overheads, etc [1, 9-12]. All of these investigations have considered noiseless (error-free) ad hoc network environments. In practice, ad hoc networks suffer from high packet-loss due to the presence of noise and node mobility, which may significantly affect the performance of this scheme. In addition, we have realized that the literature is short of clear quantitative investigations on the variation of the performance of the TSS scheme with a number of network parameters, such as nodes densities, nodes speeds, and noise-level.

The main objectives of this paper is to develop and evaluate the performance of the TSS authentication scheme in noisy MANETs and also investigate the effect of the above mentioned network parameters on the performance. The performance is evaluated by estimating the variation of the authentication success ratio against the threshold secret share for various nodes densities, nodes speeds, and network noise-levels.

The rest of this paper is organized as follows: Section 2 presents some of the most recent and related work. Section 3 describes the proposed TSS scheme. The network simulator used in this work, namely, MANSim, is briefly described in Section 4. Simulation results are presented and discussed in Section 5. Finally, in Section 6, based on the simulation results, conclusions are drawn and a number of recommendations for future work are pointed-out.

2. Previous Works

In this section we review some of the most recent work related to node authentication in MANETs. In [13], a secure group key management (GKM) scheme for hierarchical MANETs was presented, which aimed to improve both scalability and survivability of GKM for large-scale MANETs. An architectural design of mesh CA (MeCA) for wireless mesh networks (WMNs) was presented in [9]. In MeCA, the secret key and functions of CA are distributed over several mobile routers using fast verifiable share redistribution (FVSR) scheme. Simulation results showed that MeCA does not disclose its secret key even under severe attacks while incurring low overhead compared to other existing schemes in MANETs.

A lightweight authenticated key establishment scheme with privacy preservation, to secure the communications between mobile vehicles and roadside infrastructure, in a vehicular ad hoc network (VANET), was proposed in [14]. An entirely decentralized key generation mechanism was introduced in [10], in which keys can be established between group members with absolutely no prior communication. The approach relies on threshold cryptography and introduces a novel concept of node-group-key (NGK) mapping. In [15], a secure scheme for vehicular communication on VANETs was proposed. The scheme not only protects the privacy but also maintains the liability in the secure communications by using session keys.

A novel authenticated group key agreement protocol for end-to-end security in MANETs was proposed in [11]. A threshold password authentication scheme, which meets both availability and strong security requirements in MANETs, was presented in [12]. An ID-based version of the PKI cluster-based scheme was described in [16] providing secure communications in wireless ad hoc networks.

In [17], a non-interactive key agreement and progression (NIKAP) scheme for MANETs was described, which does not require an online CA. A secure and efficient key management (SEKM) framework for MANETs was presented in [18]. A novel hierarchical scheme based on threshold cryptography was proposed in [19] to address both security and efficiency issues of key management and certification service in MANET.

A fully self-organized public-key management system was proposed in [20]. It allows users to generate their public-private key pairs, to issue certificates, and to perform authentication regardless of the network partitions and without any centralized services. The applicability of threshold cryptography for membership control in peer-to-peer networks was investigated in [21]. A self-securing MANET approach was described in

[22], in which multiple nodes collaboratively provide authentication services for other nodes in the network. A design that supports ubiquitous security services for mobile hosts and it is robust against break-ins was described in [2].

3. The TSS Scheme

This section describes the overall architecture of the proposed TSS scheme for self-securing noisy MANETs.

3.1 Localized Trust Model

In the localized trust model [2, 22], an entity is trusted if any k trusted entities claim so within a certain time period T . These k entities are typically among the entity's one-hop neighbors. Once a node is trusted by its local community, it is globally accepted as a trusted node. Otherwise, a locally distrusted entity is regarded as untrustworthy in the entire network. k and T are two important parameters with T characterizing the time-varying feature of a trust relationship. Two options for setting k are as follows:

- (1) The first is to set k as a globally fixed parameter that is honored by each entity in the system. In this case, k acts as a system-wide trust threshold.
- (2) The second option is to set k as a location-dependent variable. For instance, k may be the majority of each node's neighboring nodes.

It is clear that the second option provides more flexibility to work in concert with diverse local network topology. However, there is no clear system-wide trust criterion. Due to lack of effective mechanisms to authoritatively determine a node's neighborhood in a mobile environment, the adversaries may take the advantage of this feature. In this work, we will adapt the first option with a network-wide fixed k that is tuned according to network density, network packet-loss rate (noise-level) and network robustness requirements.

3.2 The TSS Scheme

In a public-key based design, the system CA key pair is denoted as $\{SKR, SKU\}$, where SKR is the system private key and SKU is the system public key. SKR is used to sign certificates for all nodes in the network. A certificate signed by SKR can be decrypted only by the well-known public key SKU . In a TSS scheme, SKR is shared among network nodes. Each node n_i holds a secret share SKR_i , and any k of such secret share holders can collectively function as the role of CA. However, for better system security, the secrecy of SKR is preserved all the time and it is not visible, known or recoverable by any network node. Besides the system key pair,

each node n_i also holds a personal RSA key pair $\{nkr_i, nku_i\}$. To certify its personal keys, each node n_i holds the certificate C_i in the format of $\langle n_i, nku_i, T \rangle$, which reads as: "It is certified that the personal public key of n_i is nku_i during the time interval $[t, t+T]$. A certificate is valid only if it is signed by system secret key SKR .

The TSS scheme makes an extensive use of the polynomial secret sharing scheme due to Shamir [8]. A secret, specifically the certificate-signing key SKR , is shared among all n nodes in the network according to the following equation:

$$SKR(n_i) = (SKR + \sum_{j=1}^{k-1} a_j n_i^j) \bmod p \quad (1)$$

Where $SKR(n_i)$ is the node secret share, n_i is the node's ID, SKR is the system private key, k is the minimum number of shares required to recover SKR , n is the total number of nodes within the network, and p is a prime number bigger than n and SKR . In other words, the integer coefficients a_1 to a_{k-1} are either chosen between 0 and less than p ($0 \leq a_j < p$) or calculated as $a_j = a_j \bmod p$, where $i=1, 2, \dots, k$. The same is for SKR either it less than p or it is calculated as $SKR = SKR \bmod p$. A coalition of k nodes with k polynomial shares can potentially recover SKR . In fact there are two cases, these are:

- (1) A newly arrived node or a node that knows its partial share of SKR ($SKR(n_x)$), where n_x is the node ID. In this case, it needs the IDs and shares of $k-1$ nodes to construct k linear equations to solve for SKR .

- (2) A node n_x does not know its partial share $SKR(n_x)$. In this case, the node, first, needs the IDs and shares of $k-1$ nodes to calculate its share using Lagrange interpolation [23] as follows:

$$SKR(n_x) = \left(\sum_{j=1}^{k-1} SKR(n_j) \ell_{n_j}(n_x) \right) \bmod p \quad (2)$$

$$\text{where } \ell_{n_j}(n_x) = \prod_{\substack{i=1 \\ i \neq j}}^{k-1} \frac{n_x - n_i}{n_j - n_i} \quad (3)$$

Then, after having k IDs and shares, a node n_x can construct a set of k linear equations to calculate SKR . In both cases, no coalition up to $k-1$ nodes can yield any information about SKR .

3.3 The Localized Certification Procedure of the TSS Scheme

In this section, we present a description of the TSS scheme localized certification procedure for certificate issuing/renewal. In this scheme, a node n_i firstly locates a coalition B of K neighbors $\{n_1, \dots, n_K\}$ ($K \geq k$) and broadcasts certification requests to them. A node $n_j \in B$ checks its monitoring data on n_i to decide if certification service is granted, then it calculates its partial certificate and sends it back to node n_i . Upon receiving k partial certificates from coalition B , node n_i processes them together to recover its full certificate. Figure (1) outlines the main steps of localized certification procedure for the proposed TSS scheme.

// The TSS scheme localized certification procedure.
<p>For any node n_i which needs to get a new certificate or to renew its expired certificate: Locates a coalition B of K neighbors $\{n_1, \dots, n_K\}$ ($K \geq k$); Broadcasts certification requests to them; For each node $n_j \in B$ // After receiving certification request Checks its monitoring data on n_i to decide if certification is granted; If (Yes) Then Calculates its partial certificate; Sends it back to node n_i; Else (No) Discard request; End If Upon receiving k partial certificates from coalition B at node n_i: Multiplies them together to recover its new full certificate;</p>

Figure (1). The TSS scheme localized certification procedure.

There are two drawbacks in the above approach, these are:

- (1) If any node in coalition B fails to respond due to node failures or moving out of range, all the other partial certificates become useless. The computation of other nodes is all wasted and n_i has to restart the whole process from the very beginning.

- (2) When node n_j receives a certification request from n_i , its records may not provide enough information on n_i . It may be because the interaction between n_i and n_j does not last long enough. Moreover, n_i may not exist in n_j 's records at all if they just met. Node n_j has two options in this scenario. One is to serve n_i 's request, since no bad records are located.

The risk is that a roaming adversary who cannot get a new certificate from his previous location may take the advantage. The other option is to drop the request, since no records can demonstrate n_i well-behaving. But a legitimate mobile node may not be able to get a new certificate.

However, developing satisfactory solutions to the above two drawbacks is beyond the scope of this work, and may be left to future researches.

4. The MANET Simulator (MANSim)

MANSim is a computer network simulator written in C++ programming language, and it consists of four major modules: network, mobility, computational, and algorithm modules [24, 25]. The network and mobility modules were explained in [24]. Algorithm module described in Section 3. The computational module is briefly explained in this section.

Computational module: In the TSS scheme, a loop is performed over all nodes within the network to find out whether the node will be successfully authenticated. Then the number of authenticated nodes is divided by the total number of nodes within the network. This represents the success ratio. Due to the stochastic nature of the process, each node is assumed to initiate S authentication requests and the average value is calculated.

In order to consider node mobility, a simulation time is set. It is divided into a number of intervals ($nIntv$) that yields a time interval or pause time $\tau = T_{sim}/nIntv$, where T_{sim} is the total simulation time. The calculation is repeated for $nIntv$, and the results obtained for the computed parameters are averaged over $nIntv$. In general, it has been found that to obtain an adequate network performance, the pause time must be carefully chosen so that the distance traveled by the node, during location update interval, is less than the radio transmission range (R) of the nodes. Figure (2) outlines the computational modules for TSS scheme.

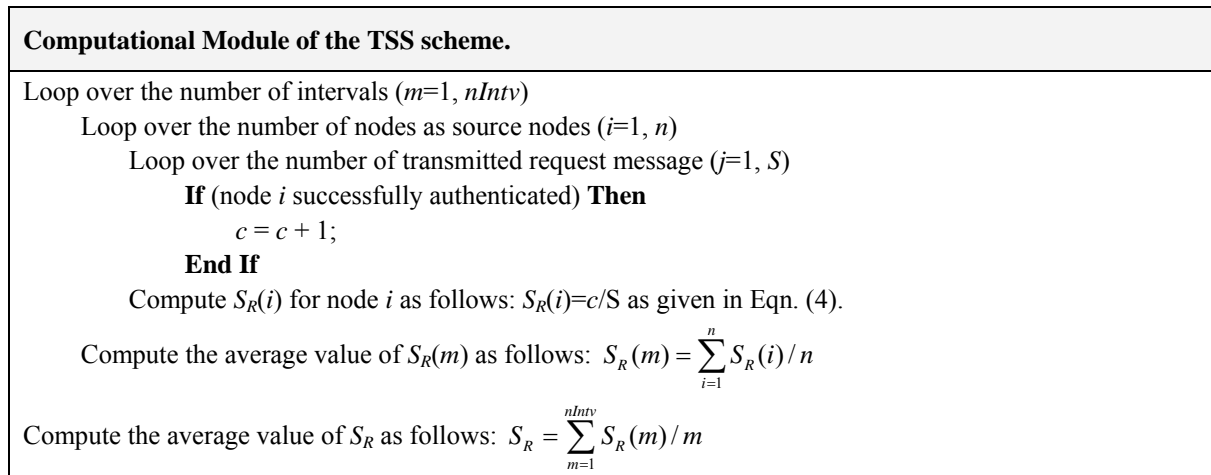


Figure (2). Computational module of the TSS scheme.

5. Results and Discussions

The performance of the proposed TSS algorithm for self-securing MANETs is evaluated in terms of a parameter known as the success ratio (S_R). S_R is defined as the ratio between the number of nodes that are successfully authenticated or certified access to the network resources (c) and the total number of nodes within the network (n). Thus, S_R can be calculated as: $S_R = c/n$; where S_R also reflects the probability with which a new arriving node can be successfully authenticated and certified access to the network resources. Using MANSim, the effect of a number of network parameters on S_R can be investigated, such as: node density (n), node mobility (u), threshold (k), and reception probability (p_c).

In order to evaluate and analyze the performance of the TSS scheme in a noisy environment, two scenarios are simulated using MANSim. These scenarios can be summarized as follows:

5.1 Scenario #1: Investigate the Effect of Node Density

Scenario #1 investigates the variation of S_R with k for various values of n . The investigations were carried-out in both noiseless and noisy MANETs environments. For a noiseless and noisy environment, p_c is taken to be 1 and 0.8, respectively. The input parameters for this scenario are given in Table (1).

Table (1) - Input parameters.		
Parameters	Scenario #1	Scenario #2
Geometrical model	Random distribution	Random distribution
Network area (A)	1000x1000 m	1000x1000 m
Number of node density (n)	100, 150, 200 nodes.	150 nodes
Transmission radius (R)	150 m	150 m
Average node speed (u)	5 m/sec	2, 5, 10 m/sec
Simulation time (T_{sim})	1800 sec	1800 sec
Threshold secret shares (k)	1, 3, 5, 7, 9, 11 nodes	1, 3, 5, 7, 9, 11 nodes
Probability of reception (p_c)	0.8 and 1.0	0.8 and 1.0
Pause time (τ)	22.5 sec	56.5, 22.5, 11.25 sec
Number of runs	20 runs	20 runs

Table (1) shows that the simulation time is 1800 sec and the pause time is 22.5 sec, which means the nodes locations are updated 80 times. Each time, S_R is calculated by dividing the number of nodes that are successfully authenticated by n . A node is considered as successfully authenticated if it establishes a link with k or more nodes from its first-hop neighbors. The values of S_R for all 80 trials are averaged to endow with the simulation S_R . Furthermore, due to the randomness of the process and to enhance the statistics of the results, each simulation is repeated for 20 runs, each run S_R is calculated, and then the average of the S_R values are calculated. The results for S_R are shown in Figure (3).

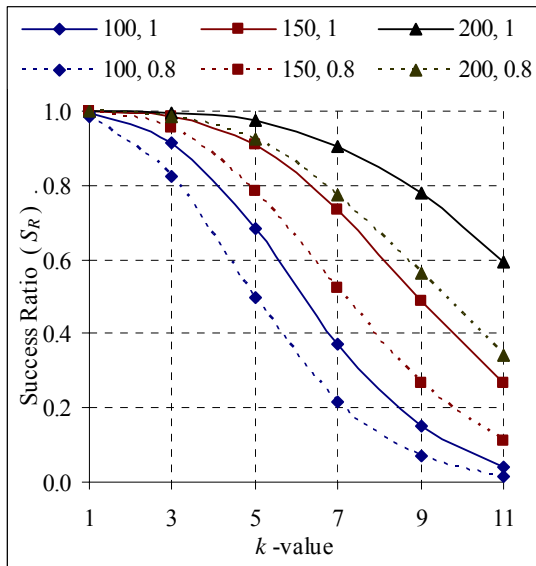


Figure (3). Variation of S_R with k for various values of n and p_c .

The main outcomes of this scenario can be summarized as follows:

- (1) As k increases, S_R nonlinearly decreases regardless of the node density for both noiseless and noisy MANETs. This is because when k increases, more first-hop neighbors are required to ensure node authentication, a case which can not be satisfied by all nodes all the time due to the randomness of nodes distribution.

- (2) For the same value of k , S_R is directly proportional to n , i.e., as n increases a higher value of S_R can be achieved. Since the node density increases the probability of having neighboring nodes equal to or higher than k nodes is most likely to happen to ensure node authentication.
- (3) For the same node density, when the noise-level increases (i.e., p_c decreases), S_R decreases. This may be explained as follows: When the node whose identity needs to be approved sends an authentication request packet asking for the secret shares of its first-hop neighbors, then due to presence of noise some of these packets may be lost or the requesting node fails to successfully receive its neighbors' replies. For example, if a node physically (distance-wise) has f_1 first-hop neighbors ($f_1 \geq k$), and due to the presence of noise some of the requests or reply packets are lost, and the node practically receives shares from f_2 nodes only ($f_2 < k$), so it can not be authenticated, and the node needs to re-initiate a new authentication request.

5.2 Scenario #2: Investigate the Effect of Node Mobility

Scenario #2 investigates the variation of S_R with k for various values of u . The investigations are carried-out in both noiseless ($p_c=1.0$) and noisy ($p_c=0.8$) MANETs. The input parameters for this scenario are given in Table (1). The simulations are carried-out in the same way explained in Section 5.1 using MANSim simulator. In this scenario three node speeds are examined, these are 2, 5, and 10 m/sec, which produce different pause times of 56.25, 22.50, and 11.25 sec, respectively. The results for the variation of S_R with k for various values of u are shown in Figure (4).

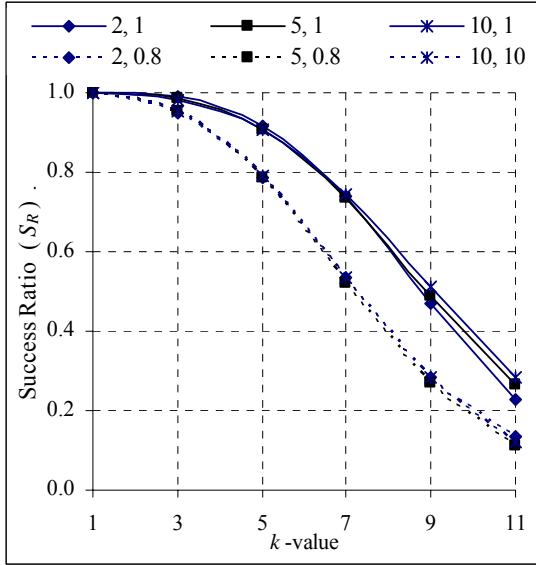


Figure (4). Variation of S_R with k for various values of u and p_c .

The results showed that u has insignificant effects on S_R . The reason for that can be explained as follows: suppose at time (t), a node distribution in which only three nodes can be authenticated out of four nodes within the network, because they have first-hop neighbors equal to or greater than 5 ($k=5$). At time $t+\tau$, the node distribution is changed where some or all nodes have randomly changed their locations. But may be still one of the nodes fails to gain access to the network resources because the number of its first-hop neighbors is less than k nodes, so that it can not be authenticated.

It can also be seen in Figure (4) that the same conclusion above is applied to both noiseless and noisy MANETs. But due to the presence of noise some of the first-hop neighbors fail to exchange their secret share with the requesting node so that the requesting node fails to gather k secret shares and it can not be authenticated. Consequently, S_R is less for noisy MANETs as compared to equivalent noiseless MANETs.

6. Conclusions

This paper demonstrated that TSS scheme is an efficient and an effective approach that can be used to provide reliable node authentication in MANETs. The security-level depends on selecting the optimum value of k which keeps a cost-effective authentication S_R , i.e., achieve node authentication with minimum delay and overhead. There are number of networks and operating parameters that affect, and should be carefully considered while, selecting an appropriate k .

The main conclusions can be summarized as follows: (1) Increasing node density has a positive effect on the security-level, since as node density increases higher k value can be selected and still achieving appropriate S_R . (2) The node speed has

insignificant effects on S_R . (3) Increasing noise-level has a negative effect on S_R , therefore, as noise-level increases it is important to reduce k to keep appropriate value for S_R .

For future work the recommendations may include:

- (1) Evaluate and investigate the variation of the performance of the TSS scheme in terms of other performance metrics, such as: load, thorough, bandwidth utilization, delay, power consumption.
- (2) Evaluate and investigate the variation of the success ratio considering the following:
 - (i) Allows nodes from the second-hop neighbors to participate in the authentication process by sending their share keys to the requesting node.
 - (ii) Instead of using a fixed k values, allows nodes to set k as a location-dependent and/or noise-level-dependent variable. For instance, k may be the majority of each node's neighboring nodes.

References

- [1] Dijiang Huang and Deep Medhi. "A Secure Group Key Management Scheme for Hierarchical Mobile Ad Hoc Networks". *Journal of Ad Hoc Networks*, Vol. 6, Issue 4, 560-577, 2008.
- [2] Jiejun Kong, Petros Zerfos, Haiyun Luo, Songwu Lu, and Lixia Zhang. "Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks". *Proceedings of the 9th International Conference on Network Protocols (ICNP '01)*, 2001.
- [3] J. Kohl and B. Neuman. "The Kerberos network authentication service (version 5)". RFC-1510, 1993.
- [4] A. Aresenault and S. Turner. "Internet X.509 Public Key Infrastructure". *Draft-IETF-PKIX-Roadmap-06.txt*, 2000.
- [5] R. Perlman. "An Overview of PKI Trust Models", *IEEE Network*, Vol. 13, 38-43, 1999.
- [6] D. Balfanz, D. K. Smetters, P. Stewart, and H. Chi Wong. "Talking to Strangers: Authentication in Ad-Hoc Wireless Networks". *Proceedings of Network and Distributed System Security Symposium 2002 (NDSS '02)*, 2002.
- [7] Seung Yi, Robin Kravets, "MOCA: Mobile Certificate Authority for Wireless Ad Hoc Networks".
- [8] Adi Shamir. "How to Share a Secret". *Communications of the ACM*, Vol. 22, No. 11, 612-613, 1979.

- [9] Jongtack Kim and Saewoong Bahk. "Design of Certification Authority Using Secret Redistribution and Multicast Routing in Wireless Mesh Networks". *Journal of Computer Networks*, in press, corrected proof, available online 10 October 2008.
- [10] A. Mukherjee, A. Gupta, and D. P. Agrawal. "Distributed Key Management for Dynamic Groups in MANETs". *Journal of Pervasive and Mobile Computing*, Vol. 4, Issue 4, 562-578, 2008.
- [11] Chan Yeob Yeun, Kyusuk Han, Duc Liem Vo, and Kwangjo Kim. "Secure Authenticated Group Key Agreement Protocol in the MANET Environment". *Journal of Information Security Technical Report*, Vol. 13, Issue 3, 158-164, 2008.
- [12] Z. Chai, Z. Cao, and R. Lu. "Threshold Password Authentication Against Guessing Attacks in Ad Hoc Networks". *Journal of Ad Hoc Networks*, Vol. 5, Issue 7, 1046-1054, 2007.
- [13] Dijiang Huang and Deep Medhi. "A Secure Group Key Management Scheme for Hierarchical Mobile Ad Hoc Networks". *Journal of Ad Hoc Networks*, Vol. 6, Issue 4, 560-577, 2008.
- [14] Chun-Ta Li, Min-Shiang Hwang, and Yen-Ping Chu. "A Secure and Efficient Communication Scheme with Authenticated Key Establishment and Privacy Preserving for Vehicular Ad Hoc Networks". *Journal of Computer Communications*, Vol. 31, Issue 12, 2803-2814, 2008.
- [15] Neng-Wen Wang, Yueh-Min Huang, and Wei-Ming Chen. "A Novel Secure Communication Scheme in Vehicular Ad Hoc Networks". *Journal of Computer Communications*, Vol. 31, Issue 12, 2827-2837, 2008.
- [16] J. S. Lee and C. C. Chang. "Secure Communications for Cluster-Based Ad Hoc Networks Using Node Identities". *Journal of Network and Computer Applications*, Vol. 30, Issue 4, 1377-1396, 2007.
- [17] Z. Li, and J. J. Garcia-Luna-Aceves. "Non-Interactive Key Establishment in Mobile Ad Hoc Networks". *Journal of Ad Hoc Networks*, Vol. 5, Issue 7, 1194-1203, 2007.
- [18] Bing Wu, Jie Wu, Eduardo B. Fernandez, Mohammad Ilyas, and Spyros Magliveras. "Secure and Efficient Key Management in Mobile Ad Hoc Networks". *Journal of Network and Computer Applications*, Vol. 30, Issue 3, 937-954, 2007.
- [19] Bo Zhu, F. Bao, Robert H. Deng, Mohan S. Kankanhalli, and Guilin Wang. "Efficient and Robust Key Management for Large Mobile Ad Hoc Networks". *Journal of Computer Networks*, Vol 48, Issue 4, 657-682, 2005.
- [20] S. Capkun, L. Buttyan, and Jean-Pierre Hubaux. "Self-Organized Public-Key Management for Mobile Ad Hoc networks". *IEEE Transactions on Mobile Computing*, Vol. 2, No. 1, 52-64, 2003.
- [21] M. Narasimha, G. Tsudik, J. H. Yi. "On the Utility of Distributed Cryptography in P2P and MANETs: the Case of Membership Control". In *IEEE ICNP*, 2003.
- [22] H. Luo, Petros Zerfos, Jiejun Kong, Songwu Lu, and Lixia Zhang. "Self-Securing Ad Hoc Wireless Networks". *Proceedings of the 7th IEEE Symposium on Computers and Communications (ISCC '02)*, 2002.
- [23] James F. Epperson. *An Introduction to Numerical Methods and Analysis*. John Wiley and Sons, 2002.
- [24] Hussein Al-Bahadili and Yousef Jaradat. "Development and Performance Analysis of a Probabilistic Flooding in Noisy Mobile Ad Hoc Networks". *Proceedings of the 1st International Conference on Digital Communications and Computer Applications (DCCA2007)*, Jordan, 1306-1316, 2007.
- [25] Hussein Al-Bahadili, Omar Al-Basheer, and Amjad Al-Thaher. "A Location-Aided Routing-Probabilistic Algorithm for Flooding Optimization in MANETs". *Proceedings of Mosharaka International Conference on Communications, Networking, and Information Technology (MIC-CNIT 2007)*, Jordan, 2007.