Trust- and Clustering-Based Authentication Services in Mobile Ad Hoc Networks

Edith C. H. Ngai and Michael R. Lyu
Department of Computer Science and Engineering
The Chinese University of Hong Kong
Shatin, Hong Kong
{chngai, lyu}@cse.cuhk.edu.hk

Abstract

A mobile ad hoc network is a kind of wireless communication network that does not rely on a fixed infrastructure and is lack of any centralized control. These characteristics make it vulnerable to security attack, so protecting the security of the network is essential. Like many distributed systems, security in ad hoc networks widely relies on the use of key management mechanisms. However, traditional key management systems are not appropriate for them. This work aims at providing a secure and distributed authentication service in ad hoc networks. We propose a secure public key authentication service based on our trust model and network model to prevent nodes from obtaining false public keys of the others when there are malicious nodes in the network. We perform an overall evaluation of our proposed approach by simulations. The experimental results indicate clear advantages of our approach in providing effective security in mobile ad hoc networks.

1. Introduction

With the advancement of wireless technology, mobile communication becomes popular in recent years. There is an increasing attention on the research of mobile distributed computing. A mobile ad hoc network is a collection of nodes with no infrastructure and these nodes are connected with wireless communication. Also, the topology of the ad hoc network is dynamically changing and the nodes of the ad hoc network are often mobile. A major challenge in the design of mobile ad hoc networks is to protect their vulnerability from security attacks. As in many distributed systems, security in ad hoc networks is based on the use of a key management system. Specific key management systems have to be developed to suit the characteristic of mobile ad hoc networks [1]. In this paper, we propose a new key management scheme with a well-defined trust model and a network model. Our trust model follows the "web of trust" approach proposed in Pretty Good Privacy [2] and we make several new contributions. Our network model is based on clustering models [3] in mobile ad hoc networks, upon which we propose a new mechanism to perform authentication. The work aims at providing a secure, scalable and distributed authentication services in the ad hoc networks.

The key features of our design are as follows. The system does not rely on any trusted-third party. Authentication can be performed in a distributed manner, and new nodes are introduced by any trustable nodes of the same group. Nodes in the network monitor the behavior of each other and update their trust tables accordingly. Our public key management mechanism endures the false certificate issued by dishonest users and malicious nodes, and avoids them to be selected as introducing nodes. These features provide a secure and highly available authentication service in the ad hoc network, which is demonstrated through our experimentation.

The remaining of this paper is organized as follows: Section 2 discusses the related work on the current key management systems developed for ad hoc networks. Section 3 formalizes the network model and the trust model, which lays the foundation for our network design, and states the system assumptions. In Section 4, we further propose the security operations on the public key certification and the update of trust tables to protect the network. The new solution is evaluated through simulation and implementation, and the results are presented in Section 5. Finally, we conclude the paper in Section 6.

2. Related work

Traditional network authentication solutions rely on physically present, trust third-party servers, or called certificate authorities (CAs). Popular network authentication architectures include X.509 standard [4] and Kerberos [5]. However, ad hoc networks are infrastructure-less, and there is no centralized server for key managements. Hence traditional solutions do not meet the requirements of mobile ad hoc networks.

Pretty Good Privacy (PGP) [2, 6] is proposed by following a web-of-trust authentication model. PGP uses digital signatures as its form of introduction. When any user signs for another user's key, he or she becomes an introducer of that key. As this process goes on, a web or trust is established. Another active research area is security function sharing [7], including a popular method for threshold secret sharing [8]. The basic idea

Proceedings of the 24th International Conference on Distributed Computing Systems Workshops (ICDCSW'04)
0-7695-2087-1/04 $20.00 © 2004 **IEEE**

is distributing the functionality of the centralized CA server among a fixed group of servers.

The paper written by Zhou and Hass [9] proposes a partially distributed certificate authority that makes use of a *(k,n)* threshold scheme to distribute the services of the certificate authority to a set of specialized server nodes. Similar to the partially-distributed CA, the fully-distributed certificate authority proposed by Luo and Lu [10] extends the idea of the partially-distributed approach by distributing the certificate services to every node. Other solutions include the self-issued certificates proposed by Hubaux et. al. [11]. It issues certificates by users themselves without the involvement of any certificate authority.

3. Models

In this section we investigate two major models related to our approach: the network model and the trust model. We survey existing work in these two models and establish the framework for our design for better security. We also state the assumptions of our system.

3.1. Primitives

As an ad hoc network is lack of infrastructure for any centralized control, its operations are usually performed in a fully distributed manner. This means every node in the network is carrying an equal role and sharing its jobs evenly. From this point of view, we perceive that the "web of trust" approach proposed by Pretty Good Privacy [2, 6] is compatible with the characteristics of the ad hoc network in providing security. An approach similar to PGP for security in mobile ad hoc networks is proposed in [11], which presents the idea of the trust graph and the method of finding a certificate chain from one user to another. However, it assumes that users are honest and do not issue false certificates, though it briefly suggests that this assumption could be relaxed by the introduction of some sort of authentication metric.

Although an authentication metric represents the assurance with which a user can obtain the authentic public key of another, it is hard to be estimated in practice. A node originally trustable to the others may become malicious or dishonest all of a sudden due to the invasion of hackers. With the above reasons, we propose a network model and a trust model to enhance the security of the public key certification in the mobile ad hoc network. The main purpose of these models is to deal with malicious nodes in a public key certification. With our clustering-based network model, behavior monitoring can be conducted in a natural way and availability is ensured for a node to find suitable introducers in the network. Our trust model employs a quantitative trust value to represent the level of trust a node holds. Trust values are stored locally to suit the distributed nature of the network. Moreover, we propose a public key certification mechanism, which enhances the discovery of malicious nodes that issue false certificates and isolate them from the future public key certificate operations.

3.2. The network model

Obtaining a hierarchical organization of a network is a well-known and well-studied problem of distributed computing. Clustering has been proven effective in minimizing the amount of storage for communication information, and in optimizing the use of network bandwidth. One class of existing clustering algorithm is based on independent dominating sets of graphs. Weight-based clustering algorithms, on the other hand, are proposed in [12]. These algorithms define a vertex with optimal weight within its neighborhood is a cluster-head, and the neighborhood of a cluster-head is a cluster. The weighting idea is generalized in [13] such that any meaningful parameter can be used as the weight to best exploit the network properties. Recent work is also performed on cluster formation such that a node is either a cluster-head or is at most *d* hops away from a cluster-head [14]. Weakly-connected dominating set is proposed for clustering ad hoc networks in [15]. A zonal algorithm for clustering ad hoc networks is proposed in [3] to divide the network into different regions and make adjustments along the borders of the regions to produce a weakly-connected dominating set of the entire graph. An adaptive method for maintaining hierarchical structure in an ad hoc network is proposed in [16], in which the role of nodes and the cluster size can be changed autonomously with the status. Finally, a model of location-aware clustering in ad hoc networks is proposed in [17]. It divides the whole network into a number of geographic zones where each zone forms a logic cluster.

Apart from the view of efficiency, we believe clustering improves the security of a network as well. Mobile ad hoc network lacks of a centralized server for management and monitoring; therefore, its security measure relies on individual nodes to monitor each other. However, direct monitoring capability is normally limited to neighboring nodes. Nodes clustering together allow the monitoring work to proceed more naturally, so as to improve the overall network security. In this paper, we propose a trust- and clustering-based public key management approach for the mobile ad hoc network. There are quite a number of existing solutions for clustering in ad hoc networks. The detailed discussion of them is beyond the scope of this paper. In our public key management approach, nevertheless, we assume the network has an algorithm to partition the nodes into different clusters with unique IDs. Figure 1 shows a mobile ad hoc network with four clusters.

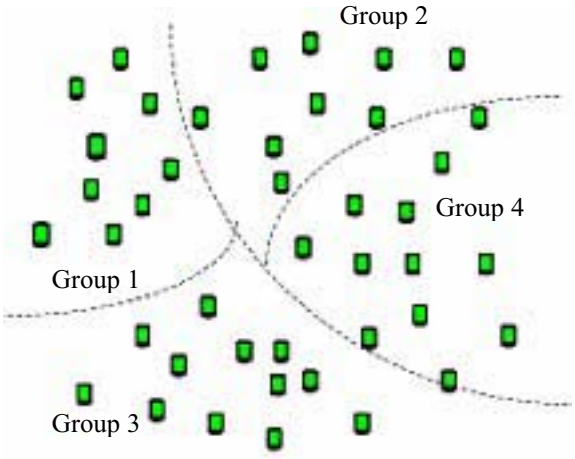

Figure 1. Clusters in mobile ad hoc network

3.3. The trust model

We define a fully-distributed trust management algorithm that is based on the web-of-trust model [18], in which any user can act as a certifying authority. The web of trust model is a cumulative trust model such that certificate might be trusted directly, or through back-tracking a chain to a directly trust root certificate, or by a group of introducers. Since our trust model does not have any trust root certificate, it just relies on direct trust and groups of introducers in certification. This model uses digital signatures as its form of introduction. Any node signs another's public key with its own private key to establish a web of trust.

Authentication in an ad hoc network without centralized certificate authorities generally depends on a path of trusted intermediaries. To evaluate the trusts from the recommendation of other reliable entities, the relying node should be able to estimate the trustworthiness of these entities. Many metrics have been proposed to evaluate the confidence afforded by different paths. One of the proposed metric represents a set of trust relationship by a directed graph [19]. It introduces the semantics of direct trust values and recommendation trust values, and shows that different values can be combined to a single value. Moreover, a metric in PGP includes three levels of trust: Complete trust, Marginal trust, and No trust [20]. Another approach explores the use of multiple paths to redundantly authenticate a channel and focuses on two notions of path independence [21].

In our trust model, we define the authentication metric as a continuous value between 0.0 and 1.0. With the consideration in our network model, we define a direct trust relationship as the trust relationship between two nodes in the same group and a recommendation trust as the trust relationship between nodes of different groups. We apply the formulae for calculation and combination of different trust values from the direct trust and the recommendation trust approach in [19].

The first formula calculates the trust value of a new path. It is a result of the computation of the direct trust values and the semantics of the recommendation values.

$$V_1 \Theta V_2 = 1 - (1 - V_2)^{V1} \quad (1)$$

The second formula is used for drawing a consistent conclusion when there are several derived trust relationships of the same trust class between two entities.

$$V_{com} = 1 - \prod_{i=1}^{m} \sqrt[ni]{\prod_{j=1}^{ni} (1 - V_{i,j})} \quad (2)$$

3.4. Assumptions

Some assumptions are made in our public key management algorithm in the mobile ad hoc network. They include:

1. Each node keeps exchanging information with other nodes on which group it belongs to.
2. Each node is able to monitor the behavior of its group mates and obtain their public keys.
3. Each node keeps a trust table for storing trust values of other nodes.

Basically, we assume that there is an underlying clustering algorithm in the network, so nodes are divided into groups with unique IDs. Nodes are equipped with some local detecting component, like watchdog for monitoring the behavior of neighboring nodes, so they can determine which nodes are trustable within the group. Finally, our trust model requires each node to keep a trust table for storing the trust values and public keys of the nodes that they know.

4. Security operations

4.1. Public key certification

Authentication in our network relies on the public key certificates signed by some trustable nodes. Let *s* be the node requesting for public key of a target node *t*. Node *s* has to ask for public key certificates signed by some introducing nodes, $i_1, i_2, ..., i_n$, as shown in Figure 2. Every node is able to request for public key certificates of any other new nodes. However, nodes in the same group are assumed to know each other by means of their monitoring components and the short distances among them. With the above assumptions, we focus on the public key certification where *s* and *t* belong to different groups. Nodes which are in the same group with *t* and which have already built up reliable trust relationship with *s* can be introducers. The introducers $i_1, i_2, ..., i_n$, reply to *s* with the public key and the trust value of *t* upon request. The trust values from $i_1, i_2, ..., i_n$, are involved in the calculation of the final trust value of *t* in *s*. Each reply message should be signed by the corresponding introducer with its private key to make it valid.

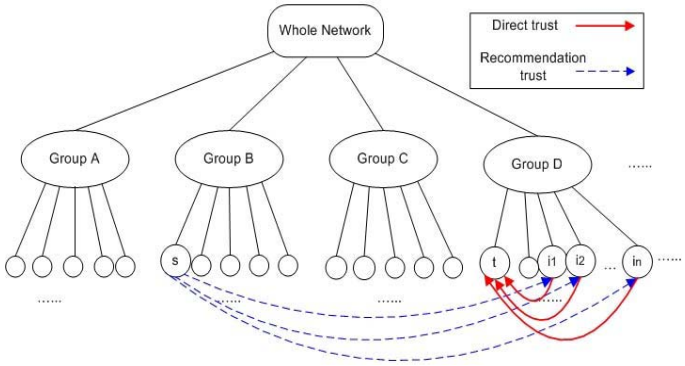

Figure 2. Public key certification

1. Looks up the group ID of t, φ_t .
2. Sorts the trust values of nodes belonging to group φ_t in the trust table. Let $i_1, i_2, \ldots, i_n \in I$, where $i_1, i_2, \ldots, i_n$ denote nodes with the highest trust values in group φ_t .
3. Sends request messages to nodes in I.
4. Collects the reply messages $m \in$ M from $i_1, i_2, \ldots, i_n$, where $m = \{Pk_t, V_{ik,t}, \ldots\}_{Sk_{ik}}$. Pk_t denotes the public key of node t, $V_{ik,t}$ denotes the trust value from i_k to t, and Sk_{ik} denotes the secret key of i_k. The reply message is signed by the secret key of i_k, Sk_{ik} .
5. Compares the public keys received and selects Pk_t with the majority votes. Let $i_{good} \in I_{good}$ and $i_{bad} \in I_{bad}$, where i_{good} are the nodes that thought to be honest (agree on Pk_t with the majority) and i_{bad} are the remaining nodes that thought to be dishonest.
6. Reduces the trust values of i_{bad} to zero. Computes and updates the trust value of t, V_t , with the following formulae:

   $V_{s,ik,t} = V_{s,ik} \Theta V_{ik,t} = 1 - (1 - V_{ik,t})^{Vs,ik}$ and

   $V_t = 1 - \prod_{k=1}^{n} (1 - V_{s,ik,t})$ where i_k denote the nodes in I_{good} and n denotes the number of nodes in I_{good} .

Table 1. Operations of *s* in public key certification

Table 1 shows the operations of s in obtaining public key certificates of t. To request the public key of t, s first looks up the group ID φ_t of node t. Then, it sorts the trust values of nodes that belong to φ_t and selects the nodes with the highest trust values as introducers $i_1, i_2, \ldots, i_n$, and sends them request messages. After collecting the reply messages that are encrypted by introducers' secret keys, s decrypts the messages with the corresponding public keys. Next, it compares the public keys obtained from the reply messages and selects Pk_t as the one with majority votes. If there is no majority vote, s tries to select more introducers and send the request messages again when it is possible. After that, it reduces the trust values of the nodes which do not agree with that public key, so to avoid selecting these nodes, now deemed dishonest or malicious, as introducers in the future. Finally, s calculates and updates the trust value of t, V_t.

4.2. Update of trust table

In Figure 3, s denotes the requesting node, and t denotes the target node, whose public key is requested by s. Nodes $i_1, i_2, \ldots, i_n$ are the introducers that reply to s with consistent public key of t. $Vs,i_1, Vs,i_2, \ldots, Vs,i_n$ denote trust values from s to introducers $i_1, i_2, \ldots, i_n$; while $Vi_1,t, Vi_2,t, \ldots, Vi_n,t$ denote trust values from introducers $i_1, i_2, \ldots, i_n$ to t. Each $V_{s,i*}$ and $V_{i*,t}$ form a pair to make up a single trust path from s to t. To compute the new trust relationship from s to t of a single path, we apply the following formula:

$$V_{s,ik,t} = V_{s,ik} \Theta V_{ik,t} = 1 - (1 - V_{ik,t})^{Vs,ik} \qquad (3)$$

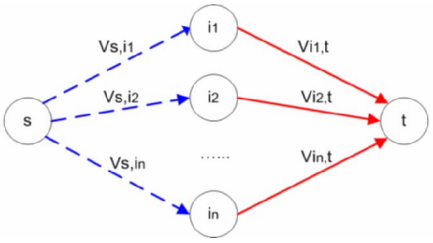

Figure 3. Trust calculation and combination

Eq. (3), extended from Eq. (1), calculates the new trust relationship from s to t via an introducer i_k. With this formula, we can calculate the n different trust values from s to t via these n introducers on different paths separately. The resulting values of $V_{s,ik,t}$ are usually different, so one has to find a way to draw a consistent conclusion. Actually, the different values do not imply a contradiction. In contrary, it can be used as collective information to compute a combined value. The following formula can be applied:

$$V_t = 1 - \prod_{k=1}^{n} (1 - V_{s,ik,t}) \qquad (4)$$

where n denotes the total number of paths.

This formula combines trust values $V_{s,ik,t}$ of different paths to give the ultimate trust value V_t of t. This final trust value V_t represents the trust value of t in the view of s after the public key certification. It contains information of trust relationships from s to different introducers, and then from different introducers to t. Finally, this value will be inserted to the trust table of s. If V_t is high, it indicates that t is trustable and it can be a possible introducer when s requests for public keys of other nodes that belong to the same group of t in the future.

5. Simulation results

We have implemented our design in network simulator Glomosim [22]. We evaluate the performance of our system in suppressing false public keys in the replies. We simulate a network that contains 40 nodes which are divided into 4 groups. Table 2 details the parameters used in our simulations. The network is assigned with a certain percentage p of trustable nodes

at initialization and a certain percentage *m* of malicious nodes. The maximum number of introducers to be selected in each request is 3. At least one introducer should give a valid reply in a successful public key certification. The simulation runs for 10000 seconds and totally 800 public key requests are sent out from different nodes. Two experiments are performed and described in the followings.

Network	# of nodes	40
	# of groups	4
	% of trustable nodes at initialization	p
	% of malicious nodes	m
Public key request	Max # of introducers for each request	3
	Min # of reply for each request	1
Simulation	Time	10000s
	# of query cycles	20
	# of requests per cycle	40

Table 2. Simulation parameters

5.1. Ratings to percentage of malicious nodes

In this experiment, we evaluate different ratings to the percentage of malicious nodes in the network with the percentage of trustable nodes to be fixed at 40% at initialization. Figure 4 shows the successful rate, failure rate, and unreachable rate on public key certification with the percentages of malicious nodes ranging from 0% to 100%. We find that the successful rate is high at the beginning and maintains over 50% until the percentage of malicious nodes increases to 80%. The failure rate keeps at a quite low level even the percentage of malicious nodes in the network is high. In the opposite, the unreachable rate can be pretty high especially when there are a lot of malicious nodes in the network. The high unreachable rate is due to the fact as most of the malicious nodes are identified, the requesting nodes cannot find any introducers to obtain the correct public keys.

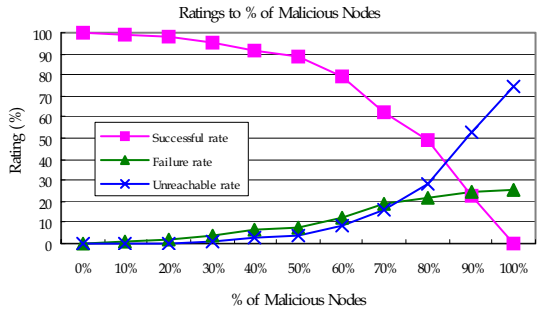

Figure 4. Ratings to percentage of malicious nodes

5.2. Comparisons among different mechanisms

In this experiment, we compare different ratings among the three public key management mechanisms in our system. These ratings, again, include the successful rate, failure rate, and unreachable rate. We fix the number of trustable nodes at initialization to be 40% and vary the percentage of malicious nodes from 0% to 100%.

The first mechanism is Pretty Good Privacy with local certificate repositories [11] in individual nodes. A user *s* verifies the public key of user *t* by finding a certificate chain from *s* to *t* in their local certificate repository. The PGP with majority vote works similarly, but it involves multiple reply messages in a request. Node *s* makes the conclusion on the public key of node *t* by majority voting. The remaining mechanism is the trust- and clustering-based algorithm proposed in this paper.

Figure 5 compares the successful rates among the three mechanisms. The two PGP mechanisms do not achieve a secure system. In these configurations, a node requests for public key certificates of another node by selecting introducers randomly, so their successful rates are low. In our trust- and clustering-based mechanism, on the other hand, each node maintains a trust table and selects introducers with high trust values. Moreover, our public key certificate mechanism can discover and isolate malicious nodes replying with false public key certificates, so it is able to maintain high successful rate.

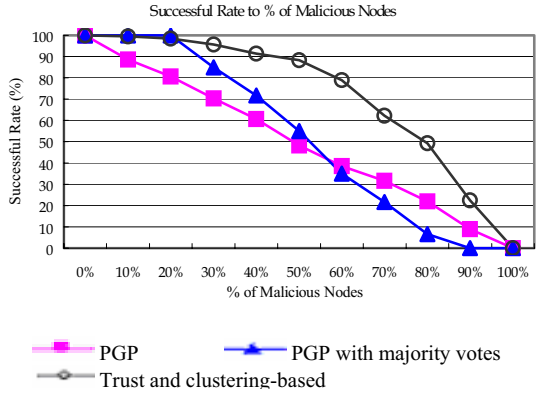

Figure 5. Comparison on successful rates

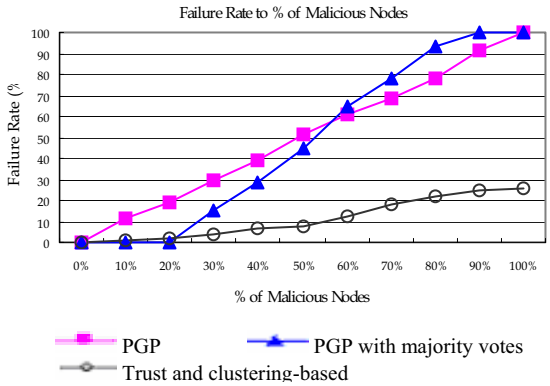

Figure 6. Comparison on failure rates

Figure 6 compares the failure rate among the three mechanisms as above. In the absence of a trustable reference for the PGP mechanisms, nodes only select

introducers randomly. Malicious nodes thus often succeed in replying false public keys; consequently, the failure rate is very high. With our trust- and clustering-based mechanism, trust values are updated from time to time for maintaining high security in public key authentication. Also, since the dishonest users issuing false certificates are located and isolated, the failure rate is kept relatively low.

6. Conclusion

This paper describes a trust- and clustering-based approach in public key authentication for mobile ad hoc wireless networks. To this end, we propose a trust model that allows nodes to monitor and rate each other with quantitative trust values. We define the network model as clustering-based, such that nodes in the network are divided into different groups with unique IDs. In this work, a trust- and clustering-based public key authentication mechanism is developed. It involves new security operations on public key certification, update of trust table, and discovery and isolation on dishonest users. In addition, we conduct the evaluation of three different approaches in public key authentication to observe their performance and characteristics in providing network security. We compare two PGP-based approaches and the trust- and clustering-based approach that we propose in this paper. With our new mechanism on public key certification, the network endures malicious nodes that issue false certificates. Our approach ensures the security and availability of public key authentication in the inherently insecure and unreliable mobile ad hoc networks.

7. Acknowledgments

The work described in this paper was fully supported by two grants, RGC Project No. CUHK4182/03E and UGC Project No. AoE/E-01/99, of the Hong Kong Special Administrative Region, China.

8. References

[1] V. Karpijoki, "Security in Ad Hoc Networks," Helsinki University of Technology, Tik-110.501 Seminar on Network Security, Telecommunications Software and Multimedia Laboratory, 2000.

[2] S.Garfinkel, "PGP: Pretty Good Privacy," O'Reilly & Associates Inc., USA 1995.

[3] Y. P. Chen and A. L. Liestman, "A Zonal Algorithm for Clustering Ad Hoc Networks," *International Journal of Foundations of Computer Science*, Vol. 14, pp. 305-322, April 2003.

[4] "Internet X.509 Public Key Infrastructure," draft-ietf-pkix-roadmap-06.txt, 2002.

[5] J. Kohl and B. Neuman, "The Kerberos network authentication service (version 5)," RFC-1510, June 1991.

[6] A. Abdul-Rahman, "The PGP trust model," *EDI-Forum: the Journal of Electronic Commerce*, April 1997.

[7] L. Gong, "Increasing Availability and Security of an Authentication Service," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 5, June 1993.

[8] T. Wu, M. Malkin, and D. Boneh, "Building Intrusion Tolerant Applications," *Eighth USENIX Security Symposium*, pp. 79-92, Washington, D.C., August 23-26 1999.

[9] L. Zhou and Z. J. Hass, "Securing Ad Hoc Networks," *IEEE Networks Magazine*, vol. 13, issue 6, 1999.

[10] J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang, "Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks," *Proceedings of the 9th International Conference on Network Protocols (ICNP)*, Riverside, California, USA, pp. 251-260, November 11-14 2001.

[11] J-P. Hubaux, L. Buttyan, and S. Capkun, "The Quest for Security in Mobile Ad Hoc Networks," *Proceedings of the 2001 ACM International Symposium on Mobile ad hoc networking & computing*, Long Beach, CA, USA, pp. 146-155, October 4-5 2001.

[12] M. Gerla and J. T. C. Tsai, "Multicluster, Mobile, Multimedia Radio Network," ACM-Baltzer Journal of Wireless Networks, vol. 1, no. 3, pp. 255-256, 1995.

[13] S. Basagni, "Distributed Clustering for Ad Hoc Networks," *Proceedings of ISPAN'99 International Symposium On Parallel Architectures, Algorithms, and Networks,* pp. 310-315, 1999.

[14] A. D. Amis, R. Prakash, T. H. P. Vuong, and D. T. Huynh, "Max-min D-cluster Formation in Wireless Ad Hoc Network," *Proceedings of IEEE INFOCOM*, March 2000.

[15] T. P. Chen and A. L. Liestman, "Approximating Minimum Size Weakly-connected Dominating Sets for Clustering Mobile Ad Hoc Networks," *The Third ACM International Symposium on Mobile Ad Hoc Networking and Computer (MobiHoc '02)*, pp. 164-172, June 2002.

[16] T. Ohta, S. Inoue, Y. Kakuda, K. Ishida, and K. Maeda, "An Adaptive Maintenance of Hierarchical Structure in Ad Hoc Networks and its Evaluation," *Proceeding of the 22nd International Conference on Distributed Computing Systems Workshops (ICDCSW '02),* 2002.

[17] J. Li and W. Jia, "Traffic Analysis in Ad Hoc Networks Based on Location-Aware Clustering," *Proceeding of the 23rd International Conference on Distributed Computing Systems Workshops (ICDCSW '03),* 2003.

[18] "How PGP Works," Chapter 1 of the document Introduction to Cryptography in the PGP 6.5.1 documentation, Copyright © 1990-1999 Network Associates, Inc. and its Affiliated Companies.

[19] T. Beth, B. Malte, and K. Birgit, "Valuation of Trust in Open Networks," *Proceedings of the Conference on Computer Security*, Sprintger-Verlag, New York, pp. 3-18, 1994.

[20] P. Zimmermann, "The Official PGP User's Guide," MIT Press, Cambridge, MA, June 1995.

[21] M. K. Reiter and S. G. Stubblebine, "Resilient Authentication using Path Independence," *IEEE Transactions on Computers* vol. 47, no. 12, pp. 1351-1362, December 1998.

[22] Xiang Zeng, Rajive Bagrodia, Mario Gerla, "GloMoSim: a Library for Parallel Simulation of Large-scale Wireless Networks," *Proceedings of the 12th Workshop on Parallel and Distributed Simulations (PADS '98)*, Banff, Alberta, Canada, May 26-29 1998.