A New Threshold Multisignature Scheme for Mobile Ad Hoc Networks

D.S.Dawoud[#1], P.Dawoud[#2] , J.Van der Merwe[#3] , C.Ndagije[#4]

[#1] Professor, Dean-Faculty of Applied Science, National University of Rwanda, Rwanda
Corresponding Author: dsdawoud@gmail.com
[#2]PhD Student, University of Waterloo, Canada,
[#3]Dr. Consultant – SA
[#4]Dr. Senior Lecturer – National University of Rwanda

ABSTRACT
Mobile ad hoc networks offer communication over a shared wireless channel without any pre-existing infrastructure. Threshold digital signatures are an important cryptographic tool used in most existing key management schemes for mobile ad hoc networks. This paper proposes a threshold-multisignature scheme designed specifically for mobile ad hoc networks. The signature scheme allows a subset of shareholders with threshold t, to sign an arbitrary message on behalf of the group. The group signature is publicly verifiable and allows any outsider to establish the identity of the individual signers. The paper proposes a self-certified public key issuing protocol that allows negotiation between a single entity and a distributed certificate authority for an implicit self-certified public key. The results of this paper will be extended and used in (Dawoud et al, 2011b), to introduce a public key management that is suitable for ad hoc networks.

Key words: Mobile ad hoc networks; pairwise key management; self-certified public key management; self-certified public key; threshold-multisignature.

1.0 INTRODUCTION

Mobile ad hoc networks by definition differentiate themselves from existing networks by the fact that they do not rely on a fixed infrastructure (Capkun et al, 2003) (Zhou & Hass, 1999). An ad hoc network of wireless nodes is a temporarily formed, spontaneous or unplanned network. Since it is created, operated and managed by the nodes themselves, ad hoc networks are solely dependent upon the cooperative and trusting nature of nodes (Buttyan & Hubaux, 2003). The self-organized and infrastructureless characteristics of mobile ad hoc networks make the design of feasible security mechanisms a challenging quest. Ad hoc network security research initially focused on secure routing protocols. All routing schemes however, neglect the crucial task of secure key management and assume pre-existence and pre-sharing of secret and/or private/public key pairs (Zhou & Hass, 1999). This left key management considerations in the ad hoc network security field as an open research area. Several proposals have already been published that provide solutions for the problem of peer-to-peer or pairwise key management in mobile ad hoc networks (Capkun et al, 2003) (Zhou & Hass, 1999) (Luo, et al, 2002) (Yi & Kravets, 2004. None of these seem to be entirely suitable for truly unplanned or impromptu mobile ad hoc networks and using them will require a tradeoff between security and the fundamental *ad hoc* characteristics. Fully self-organized ad hoc networks can be informally visualized as a group of strangers, people who have never met before, coming together for a common purpose. Since the network users are strangers, thus having no past relationships, the following issues can be identified:

1) As these strangers have to get to know each other in order to establish trust between them, bootstrapping of trust is required in the security mechanisms that protect their interests.
2) Nodes share no *a priori* keying material or a common *offline* trusted third party (TTP).
3) The identities of the nodes prior to network formation are unknown and therefore may be considered to be random or chosen by the nodes themselves as they join the network.

In ad hoc networks, to avoid a single point of vulnerability, the TTP has to be distributed (Zhou & Hass, 1999). The system secret is thus divided up into shares and securely stored by

the entities forming the distributed cryptosystem. In many existing key management proposals for mobile ad hoc networks a *distributed certificate authority* (DCA) constitutes the core of the key management services. A threshold *t* or more out of the *n* shareholders forming the DCA are required to cooperatively generate a digital signature to vouch for public key certificates. The principles of (*n, t*) threshold signature schemes and multisignature schemes are *combined* in schemes referred to as *threshold-multisignature* schemes (Hwang & Lee, 1994)) (Hwang et al, 2000) or threshold signature schemes with *traceability* (Lee & Chang, 1999) (Li et al, 2001). The threshold-multisignature schemes are based on variants of the *generalized ElGamal signatures* (Horster et al, 1994) extended to a *multiparty* setting. Threshold signature schemes with traceability guarantee the signature verifier that at least *t* members participated in the group signature and allow the signature verifier to establish the identities of the signers. Since individual signers are traceable they can be held accountable for their signatures. An important component of any threshold digital signature scheme is the sharing of the group key (Desmedt, 1994). A threshold-multisignature scheme that is suitable for ad hoc networks would require a *distributed group key generation* protocol that effectively eliminates the need for a TTP acting as a key distribution center. In threshold cryptosystems it is impractical to assume that an adversary cannot compromise more than *t* shareholders during the entire lifetime of the distributed secret (Zhou & Hass, 1999) (Herzberg et al, 1995). The secret share therefore has to be periodically updated to allow only a limited period *T* in which an *active* and *mobile* adversary must compromise a sufficient percentage of the shares in order to break the system (Herzberg et al, 1995). Also assuming the same shareholders to be present at all times is unrealistic and the availability/security tradeoff may need to be changed as a function of system vulnerability, networking environment and current functionality of the cryptosystem. This is particularly true in mobile ad hoc networks where a node may move out of transmission range, turn off due to depleted batteries or be compromised due to poor physical security. The access structure $\Gamma_P^{(n,t)}$ of the initial share distribution will thus not necessarily remain constant and will have to be redistributed to a new access structure $\Gamma_{P'}^{(n',t')}$ using a *secret redistribution protocol* (Desmedt & Jajodia, 1997) (Wong et al, 2002).cThe final goal of this research work is to introduce a new Key management protocol that takes into considerations all the above mentioned discussions. To give a complete and clear idea on our proposal, we divided the paper into two parts. In the first part, the current paper, we are introduced for the reader: a brief review for the state of the art in the field of key management for mobile ad hoc networks including the case of Vehicular ad hoc networks (Section 2); a new proposal for *threshold-multisignature* scheme which is suitable for ad hoc networks (Section 3) and ended this part of the paper by giving some comments on this proposal (Section 4). The second part of the paper (Dawoud et al, 2011b), is concentrating on a proposed cryptographic key issuing protocol called "*threshold self-certified public keying"*.

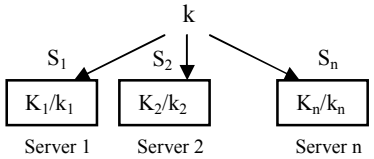

Figure 1: Key management service *K/k* configuration (Zhou & Hass, 1999)

2.0 REVIEW OF THE STATE OF THE ART

Public Key Infrastructure (PKI) provides high security services such as: key distribution, authentications, integrity and non-repudiation. Many researches attempted to establish a PKI key management solution within the constraints of ad hoc networks. The authors of (Zhou & Hass, 1999) proposed a distributed public key management service for asynchronous ad hoc networks. In Figure 1, the distributed certificate authority (DCA) consists of *n* server nodes

which, as a whole, have a public/private key pair *K/k*. The public key *K* is known to all nodes in the network, whereas the private key *k* is divided into *n* shares ($s_1, s_2, s_3, \ldots, s_n$), one for each server. The DCA signs a certificate by producing a *threshold group signature* as shown in Figure 2. Each server uses its private key share to generate a partial signature and submits the partial signature to a combiner *C*. The combiner, which can be any server, requires at least $t + 1$ shares to successfully reconstruct the digital signature. The key management service employs *proactive share refreshing* (Herzberg, 1997) to thwart mobile adversaries. The main advantage of this solution is that it addresses the lack of server infrastructure in ad hoc networks by distributing the functions of a central authority among a group of users. The major weakness of this solution (Zhou & Hass, 1999) is the fact that it assumes a common *offline* trusted third party (TTP) that distributes keying material *prior* to network formation. This also makes the solution non-scalable since all identities must be known *a priori* in order to generate and distribute the corresponding certificates.

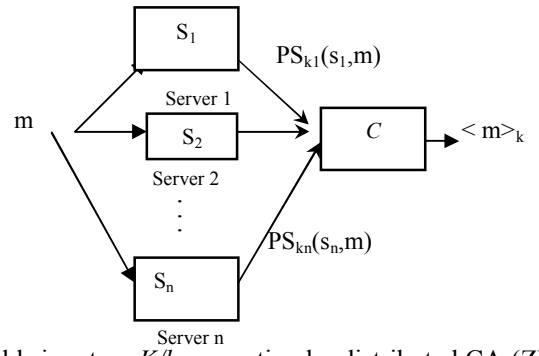

Figure 2: Threshold signature *K/k* generation by distributed CA (Zhou & Hass, 1999**)**

In (Luo, et al, 2002), the burden of holding a share of the secret key is distributed more fairly between *all* of the nodes. This increases availability since now any $t + 1$ nodes, in a local neighborhood, can renew or issue a certificate. Nodes not in possession of a share must contact at least $t + 1$ nodes to obtain a share. This proposal, as in (Zhou & Hass, 1999), suffers from the same weaknesses: a common offline TTP must distribute shares to the first $t + 1$ nodes *prior* to network formation. Since the number t is a trade off between robustness and availability the solution requires a way of adjusting t as the network expands. The solution is however not vulnerable to the Sybil attack (Douceur, 2002) as stated in (Capkun et al, 2003) since it is the responsibility of the offline TTP to ensure a one-to-one binding between a physical communication entity and its identity (certificate). The authors of (Capkun et al, 2003) proposed a public key management solution that does not require any TTP or *a priori* sharing of keying material. Each node issues its own certificates to other nodes similar to Pretty Good Privacy (PGP) (Hubaux et al, 2002). Each node keeps a limited certificate repository comprising of certificated nodes in its local neighborhood. The major drawback of (Capkun et al, 2003) is in the initialization phase. The approach takes time to setup and does not provide an answer to initial trust establishment in the network. In order for a node to sign the certificate of another node in the network an acceptable degree of trust must exist between them. By using the small world phenomena, (Hubaux et al, 2002) showed that the probability of finding a directed path between vertices of $G(V, E)$ is very high even in a very small network partition (Capkun et al, 2003). Yi *et al* (Yi & Kravets, 2004) show how a distributed certificate authority can be used in parallel with certificate chaining to eliminate some of the weaknesses of the certificate chaining approach. Yi *et al* adapted this solution from PGP.

3.0 PROPOSED DIGITAL MULTISIGNATURE SCHEME FOR MOBILE AD HOC NETWORKS

In the following we are proposing a new threshold-multisignature scheme without a trusted third party (TTP). The scheme consists of seven parts represented in the following seven subsections: subsections *3.1 to3.7*.

3.1 System Parameter Setup and Individual Self-Certificate Generation

The group members P_i, for $i = 1: n$ agree on and publish the following system parameters:

p, q	Two large primes, such that $q \mid (p-1)$.
g	Generator of the cyclic subgroup of order q in $(Z)_p^*$.
$H(\cdot)$	One-way hash function.
(n, t)	Threshold parameter t and total number of group members n.
T	Threshold cryptosystem secret update period.

Case 1*: Nonexistence of PKI or any shared information between potential protocol participants: In this case* each group member P_i, for $i = 1: n$ chooses a set of two random numbers $(SK_i, id_i) \in_R Z_q$. The member P_i uses SK_i as its private key and computes the corresponding public key as $PK_i = g^{SK_i}$. P_i also uses id_i to generate its public identity $ID_i = g^{id_i}$. P_i verifiably encrypts id_i with its public key PK_i using a *public verifiable encryption scheme* (Stadler, 1996) (Zhang & Imai, 2003) (ElGamal, 1985)(ElGamal, 1985), to generate $E_{PK_i}(id_i)$.

The main idea behind *non-interactive verifiable encryption* is that any outsider can perform a *validity test* which takes a public value (ID_i) as input and allows a prover to verify that a cipher text ($E_{PK_i}(id_i)$) encrypts id_i under a public key (PK_i) such that $(ID_i, id_i) \in R$, where R is a binary relationship ($ID_i = g^{id_i}$) (Zhang & Imai, 2003). Broadcasting the *verifiably encrypted id, i.e.,* $E_{PK_i}(id_i)$, represents a commitment to ID_i. Each P_i generates a self-certificate binding its identity ID_i to its public key PK_i. The self-certificate may be of the following structure: $SelfCert_{P_i} = PS_{SK_i}(ID_i \,||\, E_{PK_i}(id_i) \,||\, PK_i \,||\, IssueDate \,||\, CertPeriod \,||\, Ext)$. *CertPeriod* specifies the self-certificate validity period and *Ext*, optional additional extension information governed by the key issuing policy. Each P_i broadcasts its self-certificate to all P_j for $j = 1: n, j \neq i$.

Forged certificates will result in two or more certificates with the same identity. Assume a network participant receives two certificates with identity ID_i. This constitutes proof that the identity was stolen by either one of the participants. By performing a *validity test* (which takes ID_i, PK_i and $E_{PK_i}(id_i)$ as inputs) any network participant can determine which one of the two certificates is authentic. If both validity tests yield a positive result then it can be concluded that the legitimate owner of ID_i has been compromised and should reboot.

Case 2*: Case of an existing PKI:* Protocol participants P_i, for $i = 1 : n$ are assumed to have a private key SK_i and an authentic certificate, verifiable with the public key of a distributed *online* trusted third party. The certificate binds the public key $PK_i = g^{SK_i}$ to the user's identity ID_i. The certificates are distributed to all communication entities P_j, for $j = 1: n, j \neq i$.

3.2 Initial Publicly Verifiable Distributed Group Key Generation

The *publicly verifiable* distributed group key generation scheme due to Zhang *et al* (Zhang & Imai, 2003) is used to realize initial secret sharing of the group private key. The protocol is a secure, round optimal, *initial* secret sharing scheme to an access structure $\Gamma_P^{(n,t)}$ without a TTP. Note that the setup phase has been changed for the case of no existing PKI or any shared information between potential protocol participants. In this case the system parameter setup phase is as given in Section-3.1.

For each protocol participant P_i, the key generation scheme gives as output a share x_i of the group private key x_Q and the corresponding group public key $y_Q = g^{x_Q}$. P_i, $i \in Q$ is the subset of honest players after individual share verification. Each P_i can calculate its public key as $y_i = g^{x_i}$ and broadcasts the *verifiably encrypted* private key $E_{PK_i}(x_i)$ as commitment to y_i.

3.3 Individual Signature Generation

Any subset β of t or more members can represent the group and sign an arbitrary message m. Each member P_i, $i \in \beta$ selects a random integer $k_i \in (1, q-1)$ and computes $r_i = g^{k_i} \bmod p$. Each member *verifiably encrypts* k_i with its own public key PK_i using a *verifiable encryption scheme* (Zhang & Imai, 2003) (ElGamal, 1985) to generate $E_{PK_i}(k_i)$. Each member broadcasts its r_i and cipher text $E_{PK_i}(k_i)$ to all other members. This implies that each member commits to its public value r_i and provides a knowledge proof of its corresponding discrete logarithm, k_i.

After all committed r_i's are available the value R is calculated:

$$R = \prod_{i \in \beta} r_i \bmod p \tag{1}$$

Each P_i uses public values y_i and ID_i to form the sets (ID_i, y_i), for $i \in \beta$. P_i uses these sets to construct a Lagrange polynomial function $h(y)$ as follows (Li et al, 2001) (Wang et al, 1998):

$$h(y) = \sum_{i \in \beta} (ID_i \prod_{j \in \beta, j \neq i} \frac{y - y_j}{y_i - y_j}) \tag{2}$$

P_i uses secret keys x_i and k_i to compute individual signature s_i on message m from the following equation:

$$s_i = H(m, R, h(y))k_i + x_i \cdot L_\beta \bmod q \tag{3}$$

Where the Lagrange interpolating coefficient L_β is given by:

$$L_\beta = \prod_{k \in \beta, k \neq i} \frac{-ID_k}{ID_i - ID_k} \bmod q \tag{4}$$

The set (s_i, r_i) is the individual signature of P_i on message m, which is broadcast to all other group members.

Equation-3, which is used to calculate the individual signatures, is based on a variant of the ElGamal signature scheme(ElGamal, 1985) proposed by Yen *et al*. It can be shown that this variant is secure against forgery and more efficient to compute than the original ElGamal digital signature (Horster et al, 1994).

3.4 Individual Signature Verification

On receiving all the signatures (s_i, r_i), $P_j, j \in \beta$, performs the functionality of a clerk and uses the public key y_i to authenticate the individual signature of P_i by verifying whether the following equation holds:

$$g^{s_i} = r_i^{H(m,R,h(y))} y_i^{L_\beta} \bmod p \tag{5}$$

If Equation-5 fails to hold, the individual signature for message m is invalid. Participants are disqualified if their individual signatures are found to be invalid. The remaining honest participants form the set α and repeat the *individual signature generation* part from Equation-1 in Section-3.3 with $\beta = \alpha$.

3.5 Threshold Signature Generation

After $P_j, j \in \alpha$ has received and verified t or more individual signatures the group signature on message m can be obtained as (R, S) computed as:

$$R = \prod_{i \in \alpha} r_i \bmod p \tag{6}$$

$$S = \prod_{i \in \alpha} s_i \bmod q \tag{7}$$

The function $h(y)$ is attached to (R, S) and can be used later to trace the signers who collaborated to generate the threshold signature (R, S) on message m.

3.6 Threshold Signature Verification

Any outsider can use the group public key y_Q to verify the validity of the group signature (*R, S*) for a message *m* by checking if the following equation holds:

$$g^S = R^{H(m,R,h(y))} y_Q \bmod p \quad (8)$$

If Equation-8 holds, the group signature (*R, S*) for message *m* is valid.

3.7 Signer Identification

Any outsider can determine the signers of the threshold signature (*R, S*) on message *m* by using member P_i's public values (ID_i, y_i) and solve the following equation:

$$ID_i = h(y_i) \quad (9)$$

If Equation-8 and Equation-9 hold, then ID_i is a signer of the threshold signature (*R, S*) for a message *m*.

4.0 DISCUSSION AND COMMENTS ON THE PROPOSED THRESHOLD-MULTISIGNATURE SCHEME

The proposed threshold-multisignature scheme is based on a variant of the ElGamal signature scheme (ElGamal, 1985) proposed by Yen *et al* (Yen & Laih, 1993) extended to a *multiparty* setting. The scheme can equally use any other secure and efficient signature generating variant of the ElGamal signature scheme. References (Horster et al, 1994) (Harn & Xu, 1994) help in selecting such a variant. This implies that if an attack is found in future on the variant presented by Yen *et al* (Yen & Laih, 1993) used in the proposed threshold signature scheme, the variant can be replaced with a secure one.

Since the proposed *threshold-multisignature* scheme is a *natural* extension of the Yen *et al* ElGamal variant to a group setting, the security analysis is equivalent to those in (Horster et al, 1994) (Yen & Laih, 1993) and other similar threshold signature schemes (Hwang & Lee, 1994) (Hwang et al, 2000) (Li et al, 2001). The remainder of the section will only briefly consider some new attacks on similar schemes and special features of the proposed scheme.

In (Wu & Hsu, 2004), Wu *et al* present a *universal forgery* attack on the threshold signature scheme with traceable signers presented by Li *et al* in (Li et al, 2001). The attack allows adversaries to generate valid group signatures without detection. In (Wu & Hsu, 2004) it is also argued that signers in (Li et al, 2001) are in fact untraceable since the Lagrange polynomial function $h(y)$ used to identify the signers is not bounded to the group signature (*R, S*) on message *m*. This also applies to the threshold signature scheme proposed by Wang *et al* (Wang et al, 1998). A similar *framing* attack is reported by Wang *et al* in (Wang, 2002) on the threshold-multisignature signature scheme proposed by Li *et al* in (Hwang et al, 2000).

The threshold-multisignature scheme proposed in this paper eliminates universal forgery or framing attacks by using a *publicly verifiable* distributed key generation (DKG) protocol (Zhang & Imai, 2003) to construct the threshold cryptosystem and *public verifiable* encryption (Zhang & Imai, 2003) (ElGamal, 1985) to commit shareholders to their public values. Utilizing a DKG protocol also prevents adversaries from controlling the group secret key. Including the Lagrange polynomial function $h(y)$ within the *individual signatures* (Equation-3, Section-3.3) binds $h(y)$ to the threshold signature (*R, S*) on message *m*. Note that $h(y)$ in contrast to (Li et al, 2001) cannot be manipulated without detection. By evaluating $h(y)$ the identities of the signers are traceable by any outsider. A centralized combiner node is a specialized server which can be seen as a single point of vulnerability and should be distributed or replicated in ad hoc networks to ensure a correct combination (Zhou & Hass, 1999). The scheme proposed in this paper eliminates the need for a designated combiner/clerk as the construction of the threshold group signature is done by the shareholders themselves. This eliminates attacks relying on a corrupt clerk and mitigates the problem of ensuring the availability of a combiner node (Zhou & Hass, 1999). The scheme preserves the symmetric relationship between the shareholders by placing on each node the same computational, memory and communication overhead.

5.0 REFERENCES

Buttyan L., and J. P. Hubaux, 2003 "Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks," *ACM Mobile Networks and Applications*, vol. 8, no. 5, pp. 579–592.

Cagalj ,M., S. Capkun, and J. Hubaux, 2005,"Key agreement in peer-to-peer wireless networks," *Proceedings of the IEEE (Special Issue on Cryptography and Security*.

Capkun ,S., L. Buttyan, and J.-P. Hubaux, 2003, "Self-Organized Public-Key Management for Mobile Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, vol. 2, no. 1, pp. 52–64, 2003.

Dawoud ,D.S., P.Dawoud, J.Van der Merwe, C.Ndagije, 2011"A Proposed Public Key Management Scheme for Mobile Ad hoc Networks" aet2011, Uganda

Desmedt, Y.,1994, "Threshold Cryptography," *European Trans. on Telecommunications*, vol. 5, no. 4, pp. 449–457.

Desmedt, Y and S. Jajodia,1997, "Redistributing Secret Shares to New Access Structures and Its Applications," Department of Information and Software Engineering, School of Information Technology and Engineering, George Mason University, Technical Report ISSE-TR-97-01.

DouceurJ. R., 2002, "The Sybil Attack," in *proc. 1st International Workshop on Peer-to-Peer Systems (IPTPS'02)*.

ElGamal, T. , 1985,"A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. IT-31(4), pp. 469–472.

Girault, M., 1991,"Self-certified public keys," in *proc. Advances in Cryptology - EUROCRYPT'91*.

Herzberg ,A., S. Jaracki, H. Krawczyk, and M. Yung,1995 "Proactive Secret Sharing Or: How to Cope With Perpetual Leakage," in *proc. Advances in Cryptology - CRYPTO '95*.

Horster P., M. Michels, and H. Petersen, 1994,"Generalized ElGamal signatures for one message block," in *proc. 2nd Int. Workshop on IT-Security*.

Hwang , C.-M. Li, T. Hwang, and N.-Y. Lee, 1994,"Threshold-multisignature schemes where suspected forgery implies traceability of adversarial shareholders," in *proc. Advances in Cryptology - EUROCRYPT'94*.

Hwang , C.-M. Li, T. Hwang, N.-Y. Lee, and J.-J. Tsai, 2000,"(t, n) Threshold-multisignature schemes and generalized- multisignature scheme where suspected forgery implies traceability of adversarial shareholders," *Cryptologia*, vol. 24, no. 3, pp. 250–268.

Li , Z.-C., J.-M. Zhang, J. Luo, W. Song, and Y.-Q. Dai,2001, "Group-oriented (t, n) threshold digital signature schemes with traceable signers," in *proc. Topics in Electronic Commerce, Second International Symposium, ISEC 2001*.

McCune ,J. M., A. Perrig, and M. K. Reiter,2005, "Seeing-Is-Believing: Using Camera Phones for Human-Verifiable Authentication," in *proc. IEEE Symposium on Security and Privacy*.

Wang, G., 2002,"On the security of the Li-Hwang-Lee-Tsai threshold group signatures scheme," in *proc. Fifth International Conf. on Information Security and Cryptography (ICISC'02)*.

Wong , T. M., C. Wang, and J. M. Wing, 2002, "Verifiable Secret Redistribution for Archive System," in *proc. First International IEEE Security in Storage Workshop*.

Wu , T.-S. and C.-L. Hsu, 2004, "Cryptanalysis of group-oriented (t, n) threshold digital signature schemes with traceable signers," *Computer Standards and Interfaces*, vol. 26, no. 5, pp. 477–481.

Yen S.M. and C.S.Laih, 1993, "New Digital Signature Scheme based on Discrete Logarithms," Electronic Letters, vol.29, no. 12, pp. 1120-1121.

Yi ,S. and R. Kravets, 2004, "Composite Key Management for Ad Hoc Networks," in *proc. First Annual International Conf. on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous'04)*.

Yi and Kravets, 2003, "MOCA: Mobile certificate authority for wireless ad hoc networks," in *proc. of the 2nd Annual PKI Research Workshop (PKI 2003)*.

Zhou , L. and Z. J. Haas, 1999, "Securing Ad Hoc Networks," *IEEE Network: special issue on network security*, vol. 13, no. 6, pp. 24–30.