Secure Group-Based Public Key Management for Mobile Ad Hoc Networks

Eduardo da Silva, Michele N. Lima, Aldri L. dos Santos, and Luiz Carlos P. Albini

***Abstract*—Public Key Infrastructures (PKIs) perform an important role for security solutions on Mobile Ad Hoc Networks (MANETs). Changes in network paradigms and the increasing dependence of people on technology compel the development of more dependable and robust PKIs. However, designing PKIs for MANETs is a demanding task. These networks are fully distributed and have particular communication characteristics which might leave PKIs defenseless against attacks, intrusions or faults. Hence, this article presents a secure and reliable PKI for MANETs, called Secure Group-based Public Key Management (SG-PKM). SG-PKM provides a robust and secure key management service even in face of attacks or intrusions. It employs user's social behavior to assist in public key exchanges. Analytical and simulation results show the improvements obtained by SG-PKM in terms of effectiveness and survivability.**

***Index Terms*—Mobile Ad Hoc Networks, Security, Public Key, Key Management, Cryptography.**

I. Introduction

CRYPTOGRAPHIC key management is an essential service for security solutions on Mobile Ad Hoc Networks (MANETs). These networks comprise devices (nodes) communicating among themselves in a wireless multi-hop fashion, without any pre-established infrastructure or centralized control and management entity. Due to their characteristics, MANETs are prone to different threats, being cryptography in the core of security solutions.

A public key management system for MANETs, also known as public key infrastructure (PKI), must be fully decentralized and must handle different security issues [1], for example: (*i*) wireless and multi-hop communication produces opportunities for communication interceptions, interferences or eavesdropping, (*ii*) the dependence on collaborative computing allows the participation of malicious or selfish nodes in key operations. These vulnerabilities harm security attributes required for PKIs, such as confidentiality, authentication, resistance to key attacks, forward secrecy and availability [2].

Even though several key management systems for MANETs can be found in the literature [2], recent studies have demonstrated that they are ineffective in the presence of different attacks, particularly the Sybil and the lack of cooperation ones [3,4]. This work proposes a robust PKI for MANETs, called Secure Group-based Public Key Management (SG-PKM).

Manuscript received May, 2012. This work was supported in part by CAPES and CNPq (Brazil).

Eduardo da Silva, Michele Nogueira Lima, Aldri Luiz dos Santos and Luiz Carlos P. Albini are with the NR2, Informatics Department, Federal University of Paraná, Curitiba, PR, Brazil e-mail: eduardos@inf.ufpr.br, michele@inf.ufpr.br, aldri@inf.ufpr.br, albini@inf.ufpr.br.

SG-PKM is able to provide a correct key management service even in face of attacks or intrusions. SG-PKM is based on secure groups formed through trust relationship of their users, improving the secure distribution of keys and certificate generation. It also implements redundancy in PKI operations to increase the resistance and resilience against Sybil attacks. Simulation and analytical evaluation show its effectiveness and survivability to attacks.

In summary, the contributions of this article are, at least, the following ones. A robust and survivable PKI system for MANETs based on cooperation among nodes, to prove the liability users and their public keys. The formation of initiator groups based on social relationships improving the secure distribution of keys and certificate generation. The use of redundancy in many PKI operations increasing the resistance and resiliency against Sybil attacks and others.

The paper proceeds as follows. Section II discusses related work. Section III presents the models and assumptions used by SG-PKM. Section IV details SG-PKM operations. Section V depicts simulation and analytical analyses. Finally, Section VI concludes the paper and outlines future work.

II. Related Work

Plenty of public key distribution approaches for MANETs can be found in the literature. The first proposals have modified conventional key management systems to consider the wireless environment. They consider a central entity to bootstrap the service, not fitting the MANETs requirements. Self-organized key management (PGP-like) [5] is another initiative. It extends Pretty Good Privacy (PGP) [6] to implement a fully distributed certificate authority (CA). In PGP-Like, all nodes have the same role. They generate their pair of keys and issue certificates to other nodes which they trust on. Key authentication is performed through certificate chains stored in local repositories. A chain of certificates represents the trust relationship among nodes and their trustworthiness in public keys. Although PGP-like provides certain scalability, and is more suitable to the self-organization of MANETs, it is vulnerable to attacks [3],[7].

Proposals based on groups have also been designed. Some of them present characteristics such as fault-tolerance or scalability [8]-[13], being considered as robust key management initiatives. However, none of them considers attacks to the Public Key Infrastructure. Only few works have considered attacks to key management systems [14]-[17], though they are designed for wireless sensor networks. In [14]-[15], two key management schemes are presented intending to resist the node capture attacks. According to the authors, they are resistant against the attack, meaning that the adversary cannot decrypt secret communications between two

non-compromised nodes even if it compromises several nodes. In [16], an efficient key management is presented focusing on robustness and recoverability. It defines methods for distributing, maintaining and recovering session keys even in case of compromised nodes. In [17], the authors propose a key management scheme resistant to the node fabrication attacks. However, the scheme is hardware-based, being difficult for practical use.

Considering the MANETs environment, other two schemes can be found in the literature [18]-[19]. In [18], the authors propose a key management scheme for secure routing, which is robust against non-cooperative nodes. It is group-based, managing group keys and considering frequent network partitions and the absence of infrastructure. The scheme intends to be energy efficient for high key replacement rates and frequent network partitions. Each group has a round group leader which initiates the group key distribution. However, it considers only routing mechanisms.

[19] proposes the Virtual Key Management System (VKM). It is a PKI system based on a virtual structure to indicate the trust and the certificate chains formation between nodes. The virtual structure can be represented by any connected graph. In VKM, each node creates its own pair of keys. Each pair of nodes connected in the virtual structure must exchange public keys through a secure channel. After that, each node must issue certificates accordingly to the virtual structure connectivity. Key authentication is performed through certificates chain over the virtual structure. All certificates from the virtual structure must be reactively validated. If one certificate cannot be correctly verified, the entire chain is discarded. Then, the source might choose another chain or withdraw the usage of it. SG-PKM will be evaluated and analyzed against the PGP-Like and the VKM which are the two PKI schemes which best fits the MANETs environment.

III. Notation, Models and Assumptions

This section presents the network, trust and attack models used by the SG-PKM system.

Network model: SG-PKM is focused on multi-hop self-organized mobile wireless ad hoc networks, composed by n nodes ($X_1, X_2, \ldots, X_n$). There is no central control entity in the network. Indeed, all nodes have similar functionality contributing to the network maintenance, routing process and public key management. Two given nodes X_i and X_j have a physical wireless link, if their Euclidean distance is not greater than r, the communication range, and they are called neighbors. A physical path between two nodes is a set of subsequent physical wireless links. Two nodes are physically connected if there is a physical routing path starting at one node and ending at the other one.

Friendship model: Trustworthiness among nodes depends on the existing friendship of the users participating on the network. If two users, i and j, are friends, their respective devices, X_i and X_j, are also considered friends. A given node is a friend of another one only if they have exchanged their public keys. Nodes can change public key through a side channel (e.g., over an infrared channel, or pendrive) or through any key agreement protocol. Without loss of generality, in this article SG-PKM considers only bidirectional friendship between two nodes, that is, if X_i is a friend of X_j, X_j is also a friend of X_i. This assumption is based on statistical analysis of the "Web of Trust" among users of PGP. This analysis shows that about 2/3 of the links in the large strongly connected social network are bidirectional [20].

Friend relationships form a spontaneous network [21], being independent of the physical network and presenting social network properties such as small word and scale-free phenomena [22]. The former states that every pair of users can be reached through a short chain of social acquaintances [23]. The latter results from the existence of few users with a greater number of friends than others. Moreover, these few users will have high probability to be chosen by new ones as their friends, "the rich get richer" paradigm [22].

Threat model: Different types of attacks can harm PKIs on MANETs. SG-PKM is focused on those attacks which can compromise availability, confidentiality, integrity, authenticity and non-repudiation principles in a key management system. Particularly, it handles the lack of cooperation and the Sybil attacks [24]. Other attacks are out of the scope of this paper.

Lack of Cooperation: a misbehaving node, malicious or not, may stop providing authentication service as well as key storage or certificate generation, distribution or revocation. Hence, it decreases the good operation of key management services. The motivation for this attack can only be saving resources, such as storage or processing, while the node still takes part in the key management system. However, a given compromised node can maliciously participate in the key management system to damage it.

Sybil: Sybil attacks occur when adversary nodes create multiple identities in the PKI in order to manipulate keys and certificates in their advantage. In this way, false node identities can operate as legitimate ones and can, thus, violate confidentiality, authentication and non-repudiation principles.

IV. Secure Group-based Public Key Management

This section details the *Secure Group-based Public Key Management* (SG-PKM). SG-PKM is fully distributed and self-managed, and its participating nodes are organized in groups requiring no cluster heads. These characteristics contrast with those of previous proposals using cluster concepts [25]-[26]. The next subsections describe the SG-PKM operations, considering the models from Section III.

A. Overview

Initially, each node generates its pair of keys, private and public. Then, nodes must build initiator groups in order to participate in the system. An initiator group is composed of m trustful nodes meaning that all of them have mutually exchanged their public keys following existing friend relationship among their owners. The group formation capability will be evaluated and demonstrated in section V-A. Initiator groups assist the distribution of public keys and

assure the liability between public keys and node identities in a decentralized way. Initiator groups also promote redundancy. Moreover, a node can participate in different initiator groups as presented in Fig. 1. In this example, two initiator groups IG_1 and IG_2 have nodes X_4 and X_5 in common.

Each initiator group has its own pair of keys, called group key. This pair of keys can be built using any distributed key agreement scheme without a trusted third party, as the Pedersens threshold scheme (*t, m*) [27]. The group private key is used for signing digital certificates issued by its members.

Public key certificates are used to bind a public key to an identity. Hence, SG-PKM has two types of public key certificates: node certificates and group certificates. Node certificates bind node public keys with their respective identities, whereas group certificates bind group public keys with group identification. Node certificates are signed by the group in which the node participates. Group certificates are signed by another group.

It is important to point out that nodes can join or leave the networks at any time. To join the network, the new node must build a new initiator group. Note that nodes can be part of several groups. Thus, the new node must find other *m–1* friends to form a group, and connect the group into the network. On the other hand, the departure of a node from the network does not compromise its functionality. In fact, SG-PKM can tolerate up to *m–t* nodes leaving each group without interfering in its operations. If more than *m–t* nodes leave one group, the group becomes unable to perform any of its operations and is considered invalid. In this case, the remaining nodes might form a new initiator group with *m* nodes and return to the network. The limit *t* is explained in the following sections. All certificates, both node and group ones, issued by the wrecked group are valid until their expiration time, after expired these certificates cannot be revalidated.

B. Creating of node keys and forming initiator groups

For participating in SG-PKM, each node X_i individually creates its pair of keys, p_i and s_i, and finds *m–1* friend nodes to form an initiator group IG_w. Friend nodes follow the definition presented in Section III in which a node is a friend of another one only if they have exchanged their public keys through a side channel (e.g. an infrared channel) or via any key agreement protocol. Note that nodes are not allowed to join existing group, since groups must have at most *m* nodes.

Each initiator group has a unique identification to assist the generation of node and group certificates, as well as other PKI operations. Assuming that the physical network provides a unique identification for nodes, such as Media Access Control (MAC) address, the identification of a given initiator group can be defined as the hash value formed by the concatenation of all nodes identification. However, any approach can be used as long as it guarantees a unique identification for each group.

C. Generating group keys

Each initiator group must create its own pair of keys. In this way, the *m* nodes of a given initiator group IG_w cooperatively generate P_w and S_w, the public and the private keys of the group. P_w and S_w can be built using any distributed

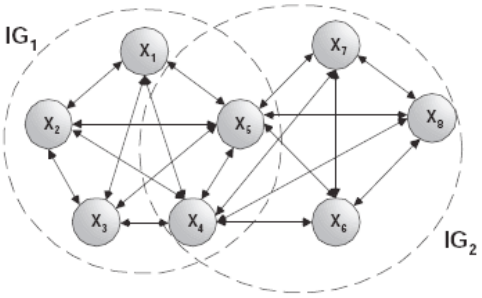


Fig. 1. Initiator Groups

key agreement scheme without a trusted third party. This work uses the Pedersens threshold scheme (*t, m*) [27]: first, a given node X_i of an initiator group IG_w randomly chooses a secret v_i belonging to Z_q. Then, it calculates $P_{w_i} = g^{v_i}$, in which:

1. q and p denote large prime numbers such that q divides $p–1$;
2. Z_q is the set of positive integers smaller than q;
3. g is an element of G_q;
4. G_q is the unique subgroup of Z_q^* of order q.

Note that each node X_i calculates its own P_{w_i}. Each P_{w_i} is sent to all *m–1* nodes in IG_w. When a node receives all *m–1* parts, it creates P_w, which is the product of the parts. As all members receive all parts, they all know P_w. Note that, while a node does receive all parts, it is not able to build the key, and consequently, it is not able to participate in the key management system.

Following the Pedersen's scheme to create the private key S_w, each node X_j randomly chooses a polynomial function $f_i(z)$ of degree (*t–1*), in which $f_i(0) = v_i$, being v_i a randomly chosen secret by X_i. Then, each node X_i calculates a $s_{ij} = f_i(j)$ for every node X_j in the group, in which $j = 1,2,3,...,m$. X_i sends s_{ij} to X_j. After receiving all m–1 subparts, X_j calculates its part of S_w by $S_{w_j} = \sum_{i=1}^{m} s_{ij}$.

Group keys are generated in a way that the public key is known by all members of the initiator group, while the private key is shared by all members using the threshold scheme (*t,m*). This means that each node in a group has a part of its private key, and at least *t* parts are necessary to build the private key. This is also the limit on the number of nodes that might leave the network without disrupting the group operations.

D. Issuing and distributing certificates

After generating P_w and S_w, each node of a given IG_w issues certificates for the public keys of the other m–1 nodes. Recalling that each node already has the public key of all other group members. These certificates are signed with S_w and locally stored. For signing certificates, S_w must be rebuilt by at least *t* members of the initiator group. By the end of this phase, all nodes in IG_w possess certificates for all other m–1 nodes. A given node certificate $C_{S_w}^j$ is composed by: an expiration time *T*, the node identity X_j, its public key p_j and the

message authentication code (MAC) of its initiator group. All this information is signed with S_w, i.e. the private key of IG_w:

$$C^{j}_{(S_w)} = (SIGN[T \| X_j \| p_j \| MAC(IG_w)]_{S_w} \| IG_w) \quad (1)$$

The public key P_w of a given initiator group IG_w also needs to be certified. Groups can issue certificates among themselves binding a given P_w with its identity. IG_z can issue a certificate $C^{IG_w}_{S_z}$ for IG_w, if IG_z "believes" in IG_w authenticity. An initiator group believes in another one if:

- at least one node in IG_z trusts two nodes in IG_w;
- two or more nodes in IG_z also participate in IG_w;

The required redundancy of two or more nodes intends to improve the reliability in evaluating public key liability. A group certificate $C^{IG_z}_{S_w}$ of group IG_z consists of:

$$C^{IG_z}_{S_w} = SIGN[T \| IG_z \| P_z]_{S_w}, \quad (2)$$

in which T is the expiration time of the group certificate, IG_z is the group identity, and P_z is its public key. All is signed with the private key of other group, in this example, S_w.

Group certificates are represented as a graph $G=(\Theta, Y)$, in which Θ represents the public keys of the groups and the set of directed edges Y represents the group certificates. Note that each vertex in the graph corresponds to a group, i.e. a set of nodes, their respective public keys and certificates. Thus, SG-PKM assumes that reaching a group certificate, it is able to reach any node certificate in the group.

E. Certificate exchange

Each node possesses two local repositories to store updated and non-updated certificates. The updated repository of a given node X_i is represented by G_i and it contains node and group certificates that are still valid. When a certificate expires, it becomes a non-updated certificate and is moved to the non-updated repository. The non-updated repository of a given node X_i is represented by G^N_i. Note that the user might impose a limit on the size of the repositories, though studies on the impact of limited repositories are out of the scope of this article.

In order to distribute the group certificates along the system, nodes periodically exchange all group certificates in their repositories with their physical neighbors. During the initiator group formation, each node knows the certificates of the groups in which it participates, and those issued by itself or by other members from its groups. With the periodic certificate exchange, nodes increase the number of group certificates in their repositories.

A given node X_i requests to its physical neighbors the list of group certificates they have within their repositories. Each neighbor responds with a message containing the hash value of group certificates in its local repositories. This message can be piggybacked on other messages of network protocols, such as those used for discovering neighbor nodes. X_i verifies which certificates it already holds and requests the neighbors for the missing ones. Hence, each neighbor sends to X_i only the missing certificates.

Certificate exchanges are performed in time intervals T_{ex}. Without loss of generality, this article assumes that all nodes have the same value of T_{ex} and that exchanges are not synchronized. Hence, if a given node X_i is sending its certificates to a node X_j, this does not mean that X_j is also sending its certificates to X_i at the same time.

F. Renewing and revoking certificates

Before node certificates have their time expired, their initiator group can issue a new version of the certificate. If t members in a given IG_y do not have any reason to revoke certificate $C^{i}_{S_y}$, they can issue an updated certificate with a new expiration time. After renewing a certificate, one copy of it is sent to all nodes of the initiator group. Group certificates can also be renewed by a subset of t nodes of the group that has originally issued the certificate. The new version of $C^{i}_{S_y}$, with the new expiration time, is sent to all nodes in the issuer group and to all nodes that have previously requested it.

Certificates can be revoked in two ways: implicitly or explicitly. Implicit revocations occur when the validity of the certificate expires. Such a revocation is automatic and local for all certificates stored in the updated repository of each node. Many reasons may cause a certificate to become invalid prior to the expiration time, e.g. changes in the relationship status between users or a suspicion that the private key is compromised. Under these situations, the certificate must be explicitly revoked. Consider that a node X_v, member of a group IG_y, wants to revoke the node certificate $C^{i}_{S_y}$ of node X_i. In this case, X_v creates a *nodeRevocation* message containing the message type and the certificate to be revoked. Then, it sends the message to all members of the group IG_y which issued the certificate. Each node of IG_y replies the *nodeRevocation* to all nodes of IG_y, only if it agrees with the certificate revocation. Once received t *nodeRevocation* messages, the members of IG_y consider $C^{i}_{S_y}$ revoked.

The revocation of a group certificate $C^{IG_w}_{S_y}$ is identical to the revocation of a node certificate, with only one exception the message type which is a *groupRevocation*. When the certificate is revoked, each node from IG_y must send the revocation message to all nodes which requested $C^{IG_y}_{S_w}$. Also, all members of IG_y keep a list L containing all nodes that had requested an update of the certificates. Such advertisements are required to inform all nodes that have the revoked certificate locally stored about the revocations.

G. Authenticating public keys

Suppose that X_i wants to authenticate the public key p_j of node X_j. Firstly, X_i requests to X_j a certificate issued to its public key. X_j replies one or more certificates issued to its public key. The number of certificates depends on the number of groups that X_j participates. Each certificate is signed with

the private key of the initiator group which issued the certificate. Then, X_i selects one of them to validate.

To validate a certificate $C^{j}_{S_w}$, X_i must use P_w, which must also be authenticated. The authentication of P_w is performed through chains of group certificates. Thus, X_i searches at least two disjoint chains of valid group certificates between any of its groups and group IG_w, in its updated repository.

If X_i does not find two chains in G_i, it merges its updated repository with the one of X_j, creating $G_\alpha = G_i \cup G_j$. Then, X_i searches again for at least two chains in G_α. If X_i does not find them in G_α, it creates $G_\beta = G_i \cup G_i^N$. In the successful case, X_i must validate the non-updated certificates used. If X_i cannot find two valid chains, it fails.

If X_i uses certificates from G_i^N or from G_j in the chains, it must validate them. To validate a certificate $C^{IG_w}_{S_y}$, X_i sends a message, called Validate Request (VREQ), to all members of the issuer group. It waits for at least *t* Validate Reply (VREP) messages from the members of the respective initiator groups. If X_i does not receive these replies in a timeout period, it is unable to validate the certificate.

V. Evaluation

The evaluation of the SG-PKM is divided in three parts: first, the analysis of its practicability considering the initiator group formation and the redundancies between groups; second, an analytical quantification of its communication cost and effectiveness; third, the demonstration of its robustness against attacks through simulations.

A. Viability of the initiator group formation

The formation of the initiator groups is a major requirement in SG-PKM. The viability analysis of having initiator groups and also redundant relationships among them is performed using the PGP database, available at http://keyring.debian.org/, and considered the methodology and metrics proposed in [28]. Initially, the PGP database is mapped into a symmetric graph $L = (W, K)$, in which W is the set of public keys representing the vertices of L, and K is the set of certificates representing its edges. Then, the L maximal cliques of different sizes were extracted. Clique in a graph means a subset of it in which any two vertices are connected by an edge. A clique is called maximal if it is not included in another clique. For SG-PKM, cliques represent initiator groups showing that all vertices (nodes) have symmetrically exchanged their public keys.

Table I presents the number of cliques and maximal cliques found in the PGP graph. The algorithms proposed in [29] were used to find the cliques. It is possible to notice that only nine vertices do not participate in any clique (groups), showing that the assumption of the initiator group is feasible.

To evaluate the redundancies in the PGP graph, graph L was transformed into a bipartite graph $L_b = (\top, \bot, K)$, in which $\top$ and $\bot$ are disjoint sets of vertices. Following [28], $\top$ represents the maximal cliques of the graph and $\bot$ is the vertices participating in cliques. There are links only between top and bottom vertices. Relating these concepts to SG-PKM, $\bot$ vertices are public keys representing their users or nodes, $\top$ vertices are initiator groups, and edges represent the participation of nodes or users into groups.

TABLE I
CLIQUE STATISTICS FOR THE PGP GRAPH WITH |W| = 956 AND |K| = 14647

Clique Size	# of Cliques	# of Maximal Cliques
> 0	293431	29070
1	956	9
2	14557	1921
3	47661	4460
4	78016	6599
5	77160	6395
6	49150	4893
9	716	351

Figure 2 depicts the distribution of vertex degree for $\top$ and $\bot$. Vertex degree is the number of neighbors of a given vertex. As observed in other social networks [28], the PGP graph follows the power law for the $\bot$ degree distribution, while the $\top$ degree distribution is Poisson shaped.

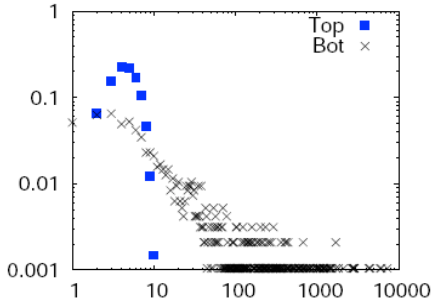


Fig. 34. Degree Distributions

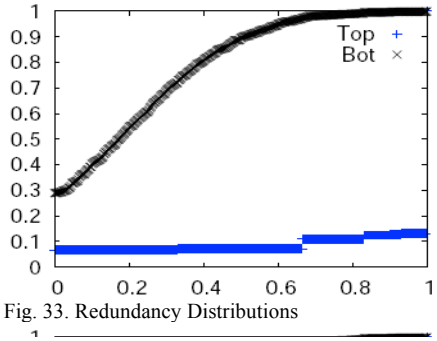


Fig. 33. Redundancy Distributions

The redundancy coefficient of a given user *v*, *rc(v)*, is used to analyze the redundancy between initiator groups in PGP, in which *rc(v)* is a fraction of pairs of neighbors of *v* linked to any other user. Being *N(v)* the set of neighbors of a user node *v*, *rc(v)* is defined in Equation 3.

Fig. 3 shows the cumulative distributions of *rc(v)* for $\top$ and $\bot$ vertices. For $\bot$ ones, 60% of them have $rc(v) > 80$, whereas 80% of them have $rc(v) \geq 50$, showing the high redundancy of the PGP graph.

$$rc(v) = \frac{|\{\{u,w\} \subseteq N(v), \exists v' \neq v, (v',u) \in K \wedge (v',w) \in K\}|}{\frac{|N(v)(N(v)-1)|}{2}} \quad (3)$$

B. Communication cost

This section contains the communication overhead analysis of the SG-PKM. Communication cost is measured in quantity of messages. The real overhead depends on the routing protocol as all messages used for authentication, revocation and renewing are exchanged through the routing protocol. Note that, the use of the routing protocol is mandatory.

In SG-PKM, when node X_i wants to authenticate the public key of a given node X_v, most operations are locally performed by X_i itself. Only if X_i does not find two valid chains in its updated repository, it will use the updated repository of X_v and its own non-updated certificate repository. Node X_i must validate each non-updated certificate and each certificate from the repository of X_v used in the chains. Thereby, the total cost to authenticate depends on the amount of certificates to be validated. The communication overhead to validate one group certificate $C_{S_y}^{IG_w}$, denoted by $VCO(C_{S_y}^{IG_w})$, is:

$$VCO(C_{S_y}^{IG_w}) = \sum_{X_j \in IG_y} (VREQ + VREP) \cdot \Delta h \quad (4)$$

in which Δh is the average number of hops between PKI nodes.

To explicitly revoke a group certificate $C_{S_y}^{IG_w}$ issued by group IG_y, each member of IG_y sends a message to all IG_y members. Further, they also send a message to all the nodes that requested an update of this certificate. Let L be the list of nodes which had requested an update of $C_{S_y}^{IG_w}$, so the communication overhead to revoke $C_{S_y}^{IG_w}$, denoted by $RCO(C_{S_y}^{IG_w})$, is:

$$RCO(C_{S_y}^{IG_w}) = \sum_{X_j \in L \cup IG_y} (\text{revoke msg}) \cdot \Delta h \quad (5)$$

in which Δh is the average number of hops between PKI nodes. Note that the total cost directly depends on the amount of nodes which had requested the validation of the certificate.

To renew a group certificate $C_{S_y}^{IG_w}$, nodes in IG_y create the new certificate version and send it to all nodes which had requested an update of this certificate and also to all thee other members of the group. Let *L* be the list of nodes which had requested an update of $C_{S_y}^{IG_w}$, the communication overhead to update/renew the group certificate $C_{S_y}^{IG_w}$, denoted by $UCO(C_{S_y}^{IG_w})$, is:

$$UCO(C_{S_y}^{IG_w}) = \sum_{X_i \in IG_y} \sum_{X_j \in L} (\text{update msg}) \cdot \Delta h \quad (6)$$

in which Δh is the average number of hops.

C. Robustness analysis

This section presents the metrics, the simulation environment and the robustness evaluation of the SG-PKM in face of lack of cooperation and Sybil attacks.

Considering two given certificates represented by their public keys, P_u and P_v, in a group certificate graph, ($p_u \rightarrow p_v$) represents a certificate chain between the two vertices. An association ($x_i \rightarrow x_j$) between two given nodes, X_i and X_j, means that X_i is able to authenticate the certificate of X_j, that is, X_i can find at least two disjoint paths connecting any initiator group of X_i with any one of X_j. Also, let *V* be the PKI node set, IG the initiator group set and $G = (\Theta, Y)$ the group certificates graph, in which Θ represents the public keys of the groups and the set of directed edges Y represents the group certificates.

The following metrics have been used to evaluate SG-PKM: Group Certificate Exchange Convergence (CE), Ratio of User Authentication (UA), Non-Compromised Group (NCG) and Non-Compromised Authentication (NCA). CE and UA are used to evaluate it in scenarios under lack of cooperation attacks, whereas NCG and NCA are used to evaluate it in scenarios under Sybil attacks. These metrics are defined as:

- CE is the average percentage of group certificates in the local repositories of the nodes at time *t*. It also represents the time needed by all nodes to obtain all groups of the PKI. CE is defined as follows:

$$CE(t) = \frac{\sum CE_i(t)}{|X|} \quad \text{in which,} \quad (7)$$

$$CE_i(t) = \frac{\sum (P_w \rightarrow P_y) \in (G_i \cup G_i^N)}{\sum (P_z \rightarrow P_x) \in G} \quad (8)$$

- UA is the average percentage of user authentications after the convergence time. User authentications are counted only if two or more disjoint certificate chains can be found to authenticate X_i. Under attack, this metric indicates the robustness of the system, evaluating if nodes will be able to authenticate other ones even in face of lack of cooperation attacks. It is defined as follows:

$$UA(t) = \frac{\sum UA_i(t)}{|X|} \quad \text{in which,} \quad (9)$$

$$UA_i(t) = \sum (X_i \rightarrow X_j) \in (G_i \cup G_i^N) \quad (10)$$

- NCG is the percentage of non-compromised groups. NCG is defined as follows:

$$NCG(t) = \frac{\sum NCG_i(t)}{|X|} \quad \text{in which,} \quad (11)$$

$$NCG_w = \begin{cases} 1 & \text{if } \neg\exists f \in IG_w : f \text{ is a false identity} \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

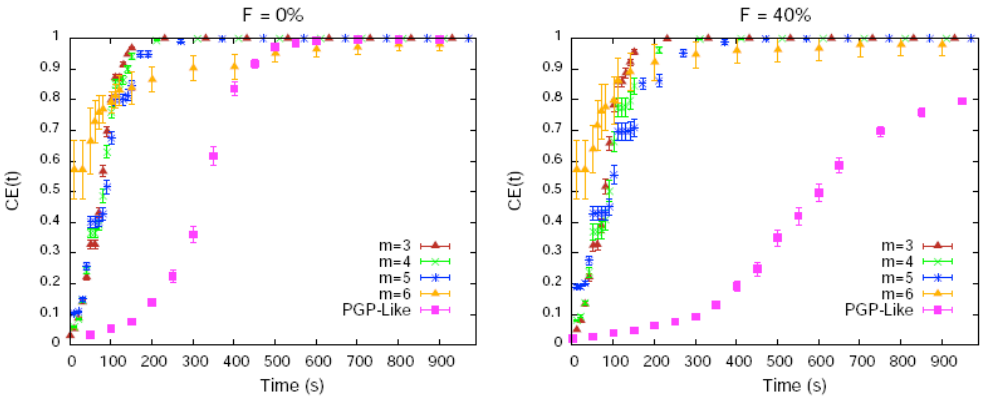


Fig. 35. Comparing convergence time of SG-PKM under Lack of Cooperation attacks

- NCA is the percentage of non-compromised public key authentications. It represents the robustness of the SG-PKM authentication process against Sybil attacks. Considering that F is the set of Sybil nodes, NCA is defined as follows:

$$NCA(t) = \frac{\sum NCA_i(t)}{|X|} \quad \text{in which,} \tag{13}$$

$$NCA_I = \begin{cases} 1 & \text{if } \neg\exists \left(P_i \rightarrow P_j\right) \ \forall f \in F \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

The Network Simulator 2 (NS-2) version 30 was used to evaluate the SG-PKM in face of lack of cooperation and Sybil attacks. Simulations were performed with 100 nodes using the IEEE 802.11 DCF as the medium access control protocol. The radio propagation follows two-ray ground propagation model and the communication range is 120m. Nodes are randomly distributed on an area of 1000 x 1000 meters and they can freely move on this area following the random waypoint model with maximal speed of 20m/s and pause time equal to 20s. The total time of simulations is 1500s and results are averages of 35 simulations with 95% confidence interval.

Public and private keys are created by nodes only during group formation. Certificates are also issued during group formation and there is no misbehavior detection mechanism in the network. Certificate exchange interval T_{ex} is 60 seconds. According to Table I, social networks present a great number of cliques with size 3, 4, 5 and 6. Thus, the evaluation of SG-PKM considers groups from 3 up to 6 nodes (the *m* value). Nodes' friendships are formed following the model proposed by Viger and Latapy [30].

Table II compares values found in PGP graphs with those from the random graphs used in the simulations. The following parameters were considered: the clustering coefficient, which is the probability of graph vertices forming a clique, the redundancy between cliques and the distance between nodes. Note that parameters on the PGP and the simulation graphs are very similar, meaning that the graphs used in the simulations present social behavior.

TABLE II.
PGP GRAPH X RANDOM GRAPHS USED IN THE SIMULATIONS

Parameters	PGP graphs	Random graphs
clustering coefficient	0.030	0.037
redundancy between cliques	0.213	0.282
distance between nodes	3.739	3.726

D. Simulation results

Initially, the effectiveness of SG-PKM is analyzed through the Group Certificate Exchange Convergence (CE) metric. Figure 4 compares the self-organized public key system proposed by Hubaux et. al [5] (called here as PGP-Like) and SG-PKM with 3, 4, 5 and 6 members into groups.

Results for two scenarios are presented, one without attackers (0% of misbehavior nodes) and another one with 40% of misbehavior nodes. In this case, misbehavior nodes issue certificates and form groups, but do not cooperate in the certificate exchange mechanism, a lack of cooperation attack.

In SG-PKM, CE reaches 100% of convergence before PGP-Like achieves it, independently of the groups size and the number of misbehavior nodes. When *m* equals to 3, 4 and 5, CE=100 before 300 seconds of network lifetime, independently of the percentage of attackers. When *m=6*, CE=95 approximately after 500 seconds of network lifetime

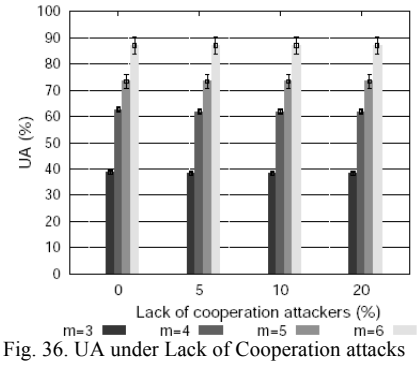


Fig. 36. UA under Lack of Cooperation attacks

when the percentage of misbehavior nodes is 0%, and after 300 seconds of network lifetime when the percentage of misbehavior node is 40%. This is a reflex of exchanging fewer certificates in total. In fact, with 0% attackers, there are 100 nodes exchanging certificates, and with 40% attackers, there

are 60 nodes exchanging certificates. Moreover, only these 60 nodes must collect all certificates.

Figure 5 shows the Ratio of User Authentication (UA) after the convergence time. Results show that increasing the group size, the percentage of user authentication also increases. For *m=5* or *m=6*, UA > 70 of valid authentications, whereas for *m=3* , it is about 40%. Furthermore, results show that UA presents the same values independently of the number of misbehaving nodes.

This behavior demonstrates the robustness of the SG-PKM to lack of cooperation attacks. However, it is important to

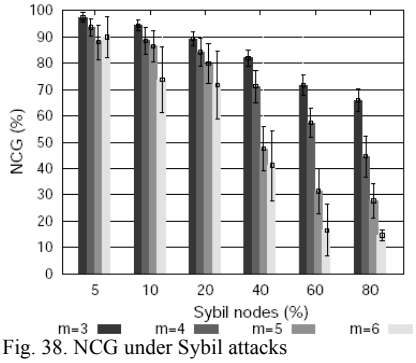


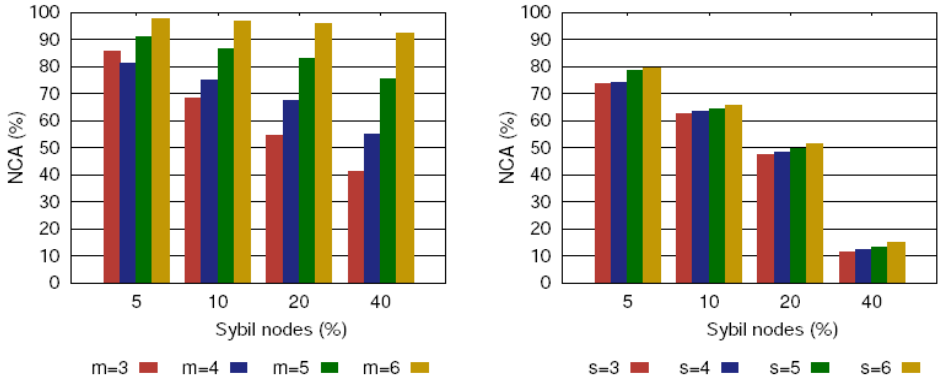
Fig. 38. NCG under Sybil attacks

point out that user authentications are calculated between non-misbehaving nodes, i.e. with 0% attackers, UA is calculated considering 100 nodes; with 20% attackers, it is calculated considering 80 nodes.

Figure 6 and Figure 7 present results related to Non-Compromised Group Ratio (NCG) and Non-Compromised Authentication Ratio (NCA), respectively. These metrics evaluate the survivability of SG-PKM in the presence of Sybil attacks. The simulated Sybil attack consists in malicious nodes creating fake nodes or impersonating authentic identities, and forming initiator groups with them. After that, they try to persuade authentic nodes to issue certificates to the false groups or nodes. Results for the PGP-Like have not been reported in the figures as it is always 100% vulnerable to the Sybil attack [3].

Results reported in Figure 6 show that with 5% of misbehavior nodes, more than 90% of the groups are not damaged, independently of the group size. For *m=3*, NCG=99. Increasing the number of misbehavior nodes to 10% and *m=3*, NCG=95. It decreases for *m=4* and *m=5*, being close to 90%. When *m=6*, NCG is still about 70%. This happens because the probability of compromising two nodes within a larger group is higher. When the percentage of misbehavior nodes is 20%, more groups are driven to issue certificates to a fake group. However, results show that the survivability of SG-PKM is still valid. In fact, with *m=6* almost 90% of the groups are not affected, for *m=4* and *m=5* it is about 85% and 80%, respectively. Only for *m=6*, NCG presents a lower value, about 70%. This happens because the probability of compromising two nodes within a larger group is higher. When the percentage of misbehavior nodes is 20%, more groups are driven to issue certificates to a fake group. However, results show that the survivability of SG-PKM is still valid. In fact, with *m=6* almost 90% of the groups are not affected, for *m=4* and *m=5* it is about 85% and 80%, respectively. Only for *m=6*, NCG presents a lower value, about 70%.

Finally, Figure 7 presents the impact of the Sybil attack on the authentication process of the SG-PKM and the VKM. Recalling, the Virtual Key Management System (VKM) is a PKI system based on a fixed virtual structure, which indicates the trust and the certificate chains formation between nodes. The virtual structure must be preloaded in each node before the network formation and it does not change during network operation. Thus, the network is not open, i.e. it is difficult to add or remove nodes from it. Each pair of nodes connected in the virtual structure must exchange public keys through a secure channel. After that, each node must issue certificates accordingly to the virtual structure connectivity. Key authentication is performed through certificates chain over the virtual structure. VKM results are reported considering a regular graph with 3, 4, 5 and 6 connections per node (the parameter *s*). SG-PKM results show that for *m=6*, the percentage of the valid nodes which correctly do not authenticate false identities is about 98% under 5% of attackers. This value is close to 97% in the presence of 10% of attackers and higher than 95% for 20% of attackers. On the other hand, with a high number of attackers (20%), the NCA presents a value lower than 70% to *m=3* and *m=4*. However, for *m=5* it is higher than 80% and for *m=6*, it is about 95%. Comparing the results of SG-PKM with the VKM ones, it is



(a) SG-PKM (b) VKM

Fig. 37. NCA under Sybil attacks for SG-PKM and VKM.

possible to see the robustness of the SG-PKM, it outperforms VKM in all scenarios, i.e. the open system of the SG-PKM is more robust against the Sybil attack than the close system of the VKM, demonstrating the validity of the proposed system.

VI. Conclusion

This article presented SG-PKM, a robust Public Key Management system for MANETs. Its goal is to provide its services and distribute keying material even in face of attacks or intrusions. It attains the survivability properties by different mechanisms, such as the formation of initiator groups based on social relationships and the use of redundancy in many PKI operations.

SG-PKM viability was evaluated considering the initiator group formation based on the PGP network. The communication overhead of the main operations were also provided, along with the survivability under lack of cooperation and Sybil attacks. Simulation results demonstrated that SG-PKM converges faster than other compared public key infrastructures. Moreover, it presents a high percentage of group certificate reachability and user authentication even in the presence of a high number of lack of cooperation attackers. In relation to the Sybil attack, even in the presence of a high percentage of fake nodes, SG-PKM is able to maintain a high percentage of non-compromised authentications.

These results demonstrate the survivability of the SG-PKM to the Sybil attack as well as to the lack of cooperation one. Facing even a very high number of attackers, it is able to correctly perform more than 70% of its operations independently of the group size, reaching almost 100% for larger groups. These results also demonstrate that the system administrator must correctly choose the adequate group size for the system. Future work includes the test of SG-PKM under different types of attacks and its association of it with a misbehavior detection mechanism. It also includes the impact analysis of having different initiator group sizes.

References

[1] Zhang, Y. Song, and Y. Fang, "Modeling secure connectivity of self-organized wireless ad hoc networks," in *IEEE INFOCOM*, 2008, pp. 251–255.

[2] V. Merwe, D. Dawoud, and S. McDonald, "A survey on peer-to-peer key management for mobile ad hoc networks," *ACM Computing Surveys*, vol. 39, no. 1, pp. 1–45, 2007.

[3] E. Silva, A. L. Santos, L. C. P. Albini, and M. N. Lima, "Quantifying misbehavior attacks against the self-organized public key management on MANETs," in *SECRYPT*, 2008, pp. 128–135.

[4] W. He, Y. Huang, R. Sathyam, K. Nahrstedt, and C. W. Lee, "SMOCK: a scalable method of cryptographic key management for mission-critical wireless ad-hoc networks," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 1, pp. 140–150, 2009.

[5] S. Capkun, L. Buttyan, and J.-P. Hubaux, "Self-organized public-key management for mobile ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 2, no. 1, pp. 52–64, 2003.

[6] P. R. Zimmermann, *The official PGP user's guide*. Cambridge, MA, USA: MIT Press, 1995.

[7] J.R.Douceur and J.S.Donath, "The sybil attack," in *Proceedings for the 1st International Workshop on Peer-to-Peer Systems (IPTPS 02)*, nov. 2002, pp. 251–260.

[8] J. Salido, L. Lazos, and R. Poovendran, "Energy and bandwidth-efficient key distribution in wireless ad hoc networks: a cross-layer approach," *IEEE/ACM Trans. on Networking*, vol. 15, no. 6, pp. 1527–1540, 2007.

[9] B. Wu, J. Wu, E. B. Fernandez, M. Ilyas, and S. Magliveras, "Secure and efficient key management in mobile ad hoc networks," *Journal of Network and Computer Applications*, vol. 30, no. 3, pp. 937–954, 2007.

[10] N. V. Vinh, M.-K. Kim, H. Jun, and N. Q. Tung, "Group-based public-key management for self-securing large mobile ad-hoc networks," in *Int. Forum on Strategic Technology (IFOST)*, 3-6 2007, pp. 250 –253.

[11] A. Poornima and B. Amberker, "A secure group key management scheme for sensor networks," in *Fifth Int. Conference on Information Technology: New Generations (ITNG)*, 7-9 2008, pp. 744 –748.

[12] R. Aparna and B. Amberker, "Key management scheme for multiple simultaneous secure group communication," in *IEEE International Conference on Internet Multimedia Services Architecture and Applications (IMSAA)*, 9-11 2009, pp. 1–6.

[13] K. Drira, H. Seba, and H. Kheddouci, "ECGK: An efficient clustering scheme for group key management in manets," *Computer Communications*, vol. 33, no. 9, pp. 1094 – 1107, 2010.

[14] K. Kifayat, M. Merabti, S. Qi, and D. Llewellyn-Jones, "Group-based key management for mobile sensor networks," in *IEEE Sarnoff Symposium*, 12-14 2010, pp. 1 –5.

[15] A. Das, "An unconditionally secure key management scheme for large-scale heterogeneous wireless sensor networks," in *First Int. Comm. Systems and Networks COMSNETS*, 5-10 2009, pp. 1–10.

[16] M. Chorzempa, J.-M. Park, and M. Eltoweissy, "Key management for long-lived sensor networks in hostile environments," *Computer Communication*, vol. 30, no. 9, pp. 1964–1979, 2007.

[17] J. Dwoskin, X. Dahai, H. Jianwei, C. Mung, and R. Lee, "Secure key management architecture against sensor-node fabrication attacks," in *IEEE Global Telecommunications Conference (GLOBECOM)*, 26-30 2007, pp. 166 –171.

[18] N. Fernandes and O. Duarte, "An efficient group key management for secure routing in ad hoc networks," in *IEEE Global Telecommunications Conference (GLOBECOM)*, nov. 2008, pp. 1–5.

[19] R. F. e Silva, E. da Silva, and L. C. P. Albini, "Resisting impersonation attacks in chaining-based public-key management on manets: the virtual public key management," in *Proc. of the International Conference on Security and Cryptography (SECRYPT 2009)*, Jul 2009, pp. 155–158.

[20] "Keyanalyze - analysis of a large OpenPGP ring," 2008, access: August 2008. [Online]. Available: http://dtype.org/keyanalyze/

[21] L. Feeney, B. Ahlgren, and A. Westerlund, "Spontaneous networking: an application oriented approach to ad hoc networking," *IEEE Communications Magazine*, vol. 39, no. 6, pp. 176–181, June 2001.

[22] L. F. Costa, F. A. Rodrigues, G. Travieso, and P. R. V. Boas, "Characterization of complex networks: A survey of measurements," *Advances In Physics*, vol. 56, pp. 167–242, 2007.

[23] J. Wu and D. J. Watts, "Small worlds: the dynamics of networks between order and randomness," *ACM SIGMOD Record*, vol. 31, no. 4, pp. 74–75, 2002.

[24] D. Djenouri, L. Khelladi, and A. N. Badache, "A survey of security issues in mobile ad hoc and sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 7, no. 4, pp. 2–28, 2005.

[25] S. Dhurandher and G. Singh, "Weight based adaptive clustering in wireless ad hoc networks," in *IEEE International Conference on Personal Wireless Communications*, January 2005, pp. 95–100.

[26] E. Ngai and M. Lyu, "Trust- and clustering-based authentication services in mobile ad hoc networks," in *International Conference on Distributed Computing Systems Workshop*, March 2004, pp. 582–587.

[27] T. P. Pedersen, "A threshold cryptosystem without a trusted party," in *EUROCRYPT*, 1991, pp. 522–526.

[28] M. Latapy, C. Magnien, and N. D. Vecchio, "Basic notions for the analysis of large two-mode networks," *Social Networks*, vol. 30, no. 1, pp. 31–48, 2008.

[29] S. Tsukiyama, M. Ide, H. Ariyoshi, and I. Shirakawa, "A new algorithm for generating all the maximal independent sets," *SIAM Journal on Computing*, vol. 6, no. 3, pp. 505–517, 1977.

[30] F. Viger and M. Latapy, "Efficient and simple generation of random simple connected graphs with prescribed degree sequence," in *Proceedings of 11th Annual International Conference of Computing and Combinatorics (COCOON 2005)*, ser. LNCS, vol. 3595. Springer, 2005, pp. 440–449.

Eduardo da Silva is a Ph.D student in Computer Science at the Federal University of Paraná. His research interests include security, wireless networks and dependability. Eduardo is also professor at Catarinense Federal Institute, Brazil. He is a Brazilian Computing Society member.

Michele Nogueira Lima is a professor at the Federal University of Paraná, Brazil. She has a Ph.D. in Computer Science by the University of Paris 6, LIP6, and her research interests include security, dependability, network management, performance modeling and wireless networks. She is a member of the IEEE Communication and Information Security Technical Committee.

Aldri Luiz dos Santos is a professor at the Department of Informatics of the Federal University of Paraná, Brazil. Aldri received his Ph.D. in Computer Science from the Federal University of Minas Gerais, Brazil. His research interests include network management, dependability, security and wireless networks. He is a member of Brazilian Computing Society and IEEE Society.

Luiz Carlos Pessoa Albini is a professor at the Department of Informatics of the Federal University of Paraná, Brazil. Luiz received his Ph.D. in Computer Science from the Informatics Department of the University of Pisa, Italy. His research interests include security, routing and energy-efficient protocols for wireless networks. He is a member of the IEEE Communication Society.