# A Sybil Safe Virtualization-based Public Key Management Scheme for Mobile Ad Hoc Networks

Renan Fischer e Silva, Eduardo da Silva, Luiz Carlos Pessoa Albini

***Abstract*— A Sybil attacker is a malicious node which creates several false identities to itself. This attack is extremely harmfull to any voting or cooperation-based system, like a MANET. MANETs (Mobile Ad hoc Networks) are dynamically estabiliched, cooperation-based wireless networks, deployed without any infrastructure or centralized administration. Due to their dynamic environment, MANETs are highly vulnerable to several malicious attacks, as the Sybil attack. Cryptography is the main technique to assure secure data transferring in these networks, making the key management an important issue. This work presents a new Key Management scheme based on virtualization, the Virtual Key Management (VKM). VKM uses a virtual structure to establish the key management operations between the nodes of the network. Therefore, nodes follow the rules established by this virtual structure to perform the issue, storage, distribution, authentication, protection and revocation of the public and certificates on network. VKM is evaluated under two different types of attacks, the personification and the Sybil, and it is also compared with two well-known key management schemes for MANETs, the PGP-Like and the GKM. VKM is the first key management for MANETs which is completely secure against the Sybil attack independently of the number of attackers and the network configuration. On the other hand, PGP-Like is completely vulnerable to a Sybil attack, and GKM becomes vulnerable with more than 40% of attackers in the network. Moreover, comparing the communication and memory overhead of these key management schemes, VKM has the smallest values independently of network configuration.**

***Keywords* — Sybil Attack, Key Management, Securitiy, Mobile Ad Hoc Networks.**

## I. INTRODUCTION

MANETs (Mobile Ad hoc Networks) are dynamically established wireless networks, which do not have any infrastructure or a centralized administration. These networks are formed by devices (nodes) which communicate through a wireless medium. Thus, nodes have a limited transmission range. To reach a destination outside its transmission range, a node forward the message to its neighboring nodes, which repeat the processes until it reaches the final destination. Therefore, routing is a crucial mechanism in such a network. A routing protocol for MANETs must be cooperative and produce routes in a distributed fashion [1,2].

MANETs inherit all security issues of the wired and wireless static networks [3]. Besides, due to the distributed routing and the wireless communication, MANETs are highly vulnerable to passive and active malicious attacks. Malicious nodes can easily intercept information (confidential or not), a passive attack, or may even interfere with the correct functioning of the network, an active attack [4,5]. Table I exemplifies several different attacks exclusively for MANETs as well as the layer where they occur. Hence, security is one of the biggest challenges in these networks.

TABLE I
TYPES OF ATTACKS IN MANETS.

| LAYER | ATTACK | DESCRIPTION |
|---|---|---|
| Physics | Noise | Interference in the transmitted signal. |
| Link | Collision | Collisions purposely caused by an attacker. |
| Network | Wormhole | Malicious nodes cooperate creating a parallel low-latency channel. |
| | Blackhole | Malicious node drops all incoming packets. |
| | Grayhole | A variation of the blackhole, the attacker selectively drops some packets. |
| | Poisoning of routing table | Malicious nodes produce messages with false routes. |
| | Blackmail | Related protocols using a black list, attackers can make false reports in order to disconnect nodes from the rest of the network. |
| Transport | Flood SYN | Flood classic TCP SYN packets, in which an attacker sends many requests to establish a connection with another node, overloading the resources of that node. |
| Multi-layers | Sybil | Malicious nodes create multiple false identities. |
| | Personification | Malicious node personificates a correct node, taking the place of a node which leaves the network or by assuming the control of the node. |
| | Lack of cooperation | Selfish nodes compromise the network operation not cooperating with their activities. |

Cryptography is considered the main technique for ensuring secure data transfers. Cryptographic systems can be classified into symmetric and asymmetric. The symmetric ones use the same key to encrypt and decrypt messages. On the other hand, the asymmetrical ones use a key to encrypt the message and another key to decrypt it [6]. The task of managing these keys

Manuscript received January 8, 2014.

Renan Fischer e Silva, Eduardo da Silva and Luiz Carlos Pessoa Albini are with NR2 − Informatics Department, Federal University of Paraná, Curitiba, Brazil. E-mail: {renan, eduardos, albini}@inf.ufpr.br

Corresponding Author: Luiz Carlos Pessoa Albini – albini@inf.ufpr.br

is called key management. A key management scheme should define correct procedures to issue, store, distribute, protect and revoke keys [7].

Key management in MANETs [7, 8, 9, 10, 18, 19, 20, 21] must also consider the dynamic topology, and should be self-organized and decentralized [7]. They can be classified in [8]: identity-based, based on certificate chains, based on clusters, based on pre-distribution and based on mobility. Among these, the ones based on certificate chains seems to fit best in the MANETs environments, once they are fully distributed and self-organized and do not require any type of central entity. The main scheme based on certificate chains is the *Self-Organized Public Key Management System* [9,10], called in this work PGP-Like.

In PGP-Like, keys are authenticated through chains of certificates, using local certificate repositories. However, the use of certificate chains makes it highly vulnerable to Sybil and personification attacks. In fact, after 800 seconds of network lifetime, more than 80% of all chains are contaminated by malicious nodes. Consequently, more than 80% of all chains are invalid.

The first key management for MANETs which provides some security against personification attacks is the *Group-based Key Management* (called in this work GKM) [12]. Authors demonstrate that GKM is more resistant to Sybil and personification attacks than PGP-Like. In fact, it is able to correctly perform more than 90% of all authentications. However, this number depends on the size of the groups. Larger groups provide better resistance to the attacks. On the other hand, larger groups imply in larger overheads to groups management.

This work presents a new Key Management System based on Virtualization, called Virtual Key Management System (VKM). VKM uses a virtual network, or virtual structure, to establish the certificate issue mechanism. Moreover, nodes follow the virtual network to issue, store and revoke all certificates as well as to validate all public keys and certificates in the network. A preliminary version of VKM can be found in [13]. However, it does not describe all operations of VKM and does not present the results for the Sybil attack.

VKM is highly resistant to the Sybil attacks. In fact, a Sybil attack has no effect over it, independently of the amount of malicious nodes. When considering a personification attack, VKM performs more than 80% of all authentications for 5% of attackers, but this number decreases as the number of attackers increases.

As GKM and VKM are the main virtualization-based key management for MANETs which provide security against the Sybil attack, this article also provides a comparison study between them. This study considers their ability to resist attacks as well as their memory and communication overhead. Thus, this study can help network managers on choosing the correct key management scheme for their network.

The rest of this article is organized as follows: Section 2 describes PGP-Like and GKM; Section 3 presents the VKM; Section 4 contains the VKM analysis and a comparison between the three schemes when submitted to the Sybil and the personification attacks. Finally, Section 5 presents the conclusions of this work.

## II. Related Work

This section presents the *Self-Organized Public Key Management System* (PGP-Like) [9,10] and the *Group-based Key Management* (GKM) [12], which are two of the main key management schemes for MANETs.

### A. Self-Organized Public Key Management System

The *Self-Organized Public Key Management System* (PGP-Like) is a fully distributed and self-organized public key management scheme for MANETs [9,10]. Nodes running PGP-Like create their own public and private keys following the PGP concepts [11].

In PGP-Like, if a node $u$ believes that a given key public $K_v$ belongs to node $v$, it can issue a certificate tying $K_v$ to $v$. This certificate is stored in the local certificate repository of $u$ and $v$. The local repository is represented by a directed graph $L=(D,E)$, where $D$ represents the public keys and $E$ the certificates. Therefore, an edge between two vertices $K_u$ and $K_w$, $K_u \rightarrow K_w$ denotes a certificate, signed by $u$, tying $K_w$ to node $w$. A path connecting two vertices $K_u$ and $K_w$ is represented by $K_u$ a $K_w$.

Each node $u$ maintains two local certificate repositories, the updated $L_u$ and the non-updated $L_u^N$ ones [9]. The non-updated certificate repository contains the expired certificates. Nodes exchange their repositories with their local physical neighbors in regular time intervals.

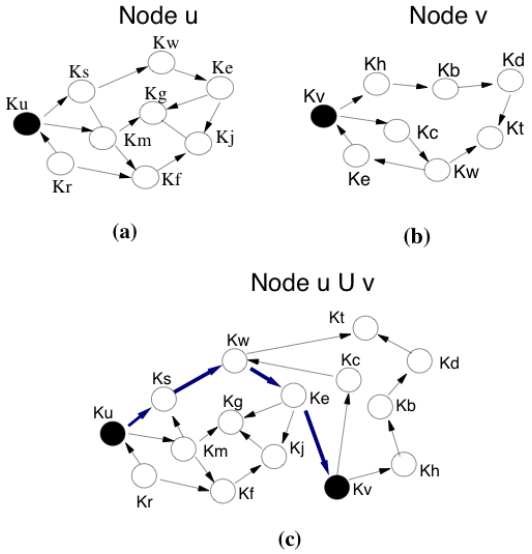


Figure 1. Certificate repositories (a) and (b); a certificate chain (c).

When node $u$ wants to authenticate the public key $K_v$ from node $v$, it first tries to find a path from $K_u$ to $K_v$ on

$L_u$. If $\exists K_u \mapsto K_v \in L_u$, node $u$ authenticates $K_v$. If $\neg\exists K_u \mapsto K_v \in L_u$, node $u$ creates, $L_1 = L_u \cup L_v$, and tries to find $K_u \mapsto K_v \in L_1$. If such a path exists, the authentication succeeds. If $\neg\exists K_u \mapsto K_v \in L_1$, then node $u$ creates $L_2 = L_u \cup L_u^N$, and tries to find $K_u \mapsto K_v \in L_2$. If $\exists K_u \mapsto K_v \in L_2$, node $u$ must validate all certificates from $L_u^N$ used in $K_u \mapsto K_v \in L_2$. If $\neg\exists K_u \mapsto K_v \in L_2$, then the node $u$ is not able to authenticate $K_v$.

Paths found in the repositories are certificate chains. Certificate chains are considered weak endorsements because they are transitive. Unfortunately, ensuring a valid transitive trust between two nodes is very difficult [14]. For this reason, if any node in the chain is compromised, the entire chain is considered compromised, providing a false authentication.

### B. Group-based Key Management

The *Group-based Key Management* (GKM) [12] is a virtualization-based scheme based on groups. To participate in the key management scheme, users must form small groups, called *initiator groups* ($IG$), in which all nodes have the same role without leaders. These groups are supposed to be formed based on the relationship of the users and they form a virtual group network. The virtual group network is used to support all key management operations, such as authentication, revocation and update of keys.

In GKM, each node $u$ creates its own pair of keys, public ($K_u$) and private ($prK_u$). Then, it needs to find other $m-1$ trusted nodes and, with these nodes, form a group $IG_w$. Nodes in an $IG_w$ exchange their public keys among themselves using a secure channel. Then, they generate a pair of keys for the group, a public ($GK_w$) and a private ($prGK_w$) group key, following any key agreement scheme. After generating $GK_w$ and $prGK_w$, each node in $IG_w$ issues certificates for the public keys for the other $m-1$ nodes of the group. These certificates are signed with $prGK_w$ and locally stored. At the end of this phase, all nodes in $IG_w$ have certificates for all nodes in the group. These are called *node certificates*.

Group public key $GK_w$ must also be certified. Groups can issue certificates among themselves binding a given $GK_w$ with its identity $IG_w$. In other words, $IG_z$ can issue a certificate biding $GK_w$ with $IG_w$ if it *believes* $IG_w$. In GKM, a group believes another one if at least one node in two or more nodes participate in both groups. These certificates are called *group certificates*.

Each node possesses four local repositories to store certificates: two for updated certificates, group and node ones, and two for non-updated certificates, group or node ones. Updated repositories contain valid certificates, while the non-updated repositories contain the expired ones. Nodes periodically exchange their group certificates within their physical neighbors. During group formation, each node knows only the certificates of the groups in which it participates, and the certificates that the nodes in its group have issued for other groups. With the periodic certificate exchange, nodes increase the number of group certificates in their repositories.

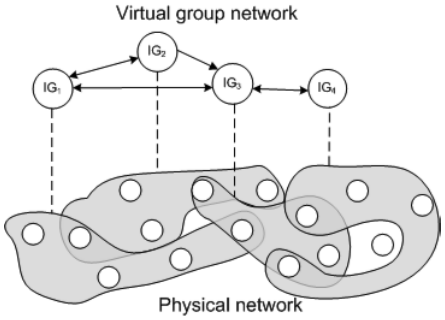


Figure 2. Group Network of GKM.

Each group certificate repository can be represented by a group certificate graph $G=(V,E)$, in which the set of vertices $V$ represents the public keys of the groups, and the set of directed edges $E$ represents the issued group certificates. When a node $i$ wants to authenticate the certificate of a node $j$, it must use the group public key $GK_w$ for any $w$ such that $j \in w$. However, before using a group public key, node $i$ must authenticate it.

The authentication of $GK_w$ is performed through a chain of group certificates. Node $i$ searches for at least two chains of valid group certificates between its group and $IG_w$ in its updated group certificate repository. If $\exists\, GK_\alpha \;\mathsf{a}\; GK_w \in G_i \therefore i \in IG_\alpha$, node $i$ can validate $GK_w$ and, then, it can validate the certificate of $j$. If $\neg\exists\, GK_\alpha \;\mathsf{a}\; GK_w \in G_i \therefore i \in IG_\alpha$, node $i$ merges its updated group repository with the one of $j$ ($G_1 = G_i \cup G_j$). Then, $i$ tries to find at least two chains of group certificates in $G_1$. If even after merging the repositories, $i$ does not find the chains, it will try to find them in the union of its updated and non-updated repositories $G_2 = G_i \cup G_i^N$. If it succeeds in finding two chains in $G_2$, it must validate all certificates from $G_i^N$. If it is not able to find such chains, it will not be able to authenticate the certificate of node $j$.

Certificate update and revocation are performed following a process similar to the group key agreement. This implies a high communication overhead, as all operations are based on group agreements. Note that even though larger groups improve security, as it will be shown in section 4, they also impose higher overheads and delays on group operations.

## III. VIRTUAL KEY MANAGEMENT

The *Virtual Key Management* (VKM) is a virtualization-based scheme, which uses a virtual structure to define all key operations and to assist the formation of certificate chains. The virtual structure is represented by a directed graph $H=(W,VL)$, which is not related to the current topology of

the network. The set $W$ represents the nodes and the set $VL$ represents the virtual links. A virtual link $(u,v) \in VL$ indicates that the node $u$ issues a certificate binding $K_v$ to node $v$. The virtual structure must be preloaded in each node during network initialization. In VKM, each node $u$ must issue a certificate to each node with which it has a directed connection in the virtual structure, unless node $u$ discovers that a node has malicious behavior. Before node $u$ issues a certificate to node $v$, node $u$ must obtain the public key of $v$ through a secure channel, such as infrared or smart cards, before the network formation or through a key agreement protocol. The evaluation of this parameter is outside the scope of this article.

VKM is independent of graph used as virtual structure, though the graph should be regular to assure a traffic distribution between all nodes. The most appropriate virtual structure should be selected by the user considering properties such as diameter, bisection width and scalability. Moreover, all nodes must use the same virtual structure, though they do not keep updated information about the certificates, thereby reducing the memory used by local repositories. This work uses *Ring of Rings* (Figure 3) as the virtual structure.

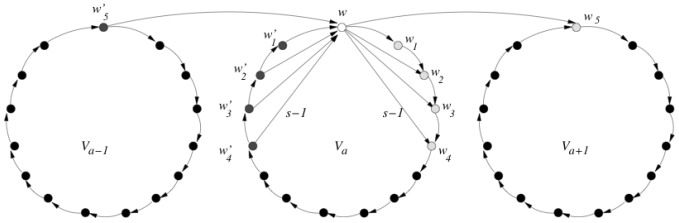


Figure 3. Virtual structure RoR.

Figure 3 illustrates the graph *Ring of Rings* (RoR) [15], with 45 nodes split in three rings of 15 nodes. Each node has a direct connection to five other nodes, meaning they are responsible for issuing five certificates. For example, node $w$ is responsible for issuing certificates tying $K_{w_1}$ to $w_1$, $K_{w_2}$ to $w_2$, $K_{w_3}$ to $w_3$, $K_{w_4}$ to $w_4$ and $K_{w_5}$ to $w_5$. Furthermore, nodes $w'_1$, $w'_2$, $w'_3$, $w'_4$ and $w'_5$ are responsible for issuing certificates tying $K_w$ to $w$.

All certificates are issued with a limited lifetime $T_j$ and after $T_j$ it is considered expired. Before the certificate expires, the issuer can update it, by issuing a new version with an extended lifetime $T'_j$.

When a certificate is issued, the issuer stores it in its local repository and sends a copy for the corresponding node, which also stores it. Nodes store only the certificates they issued and certificates that were issued to them.

To authenticate the public key of a node $j$, node $i$ needs to find a path from itself to $j$ in the virtual structure, a virtual path. Virtual paths are certificate chains. Note that it might be possible to find several virtual paths from $i$ to $j$, node $i$ can choose any of them, or even try more than one at a time. After choosing a virtual path, the source must obtain all certificates from the nodes in the virtual path, i.e. it must validate the entire certificate chain. The certificate validation process is performed as follows:

1. the first certificate is directly verified by node $i$, as it is the issuer of this certificate;
2. each remaining certificate is verified using the public key of the previous certificate;
3. the last certificate contains the public key of node $j$.

VKM guarantees that only correct and valid certificates are used. However, it implies in endorsement latency, as certificates must be reactively validated. If the network uses a virtualization-based routing protocol, such as VRP [15] or VDV [16], VKM can use the same virtual network graph, reducing the memory usage even more.

Certificate revocation can be explicit or implicit. Implicit revocations are based on the certificate lifetime. If a certificate issuer does not update the certificate, it is considered revoked. In the explicit revocation, if a node $u$ believes that another one $m$ is presenting malicious behavior, $u$ contacts all nodes which issue a certificate to $m$ accusing it of misbehavior. The issuers start a voting mechanism to decide if they believe the accusation or not. If they believe the accusation, all issuers revoke node $m$ certificates. If they do not believe it, they accuse $u$ of misbehavior.

Like GKM, the overhead to revoke a certificate depends on the number of issuers, i.e. the connectivity of the virtual structure. As it will be shown in section 4, the security of the network depends on the connectivity of the virtual structure. Thus, the network manager must choose the correct values balancing security and overhead, based on the network requirements.

## IV. RESULTS

This section presents the evaluation of VKM, its effectiveness against the personification and Sybil attacks and a comparison with the PGP-Like and the GKM. All evaluations were performed through simulations on the *Network Simulator 2* (NS-2) [17], version 30. Simulation parameters are shown in Table II and they are the same used on the original evaluations of PGP-Like and GKM. All results are averages of thirty-five simulations with a confidence interval of 95%. GKM considers 50 overlapped virtual groups randomly formed with 6, 9 and 12 members. The virtual structure of VKM is the RoR with four rings and twenty-five nodes per ring. Each node issues 5, 10, 15 and 20 certificates, and has 5, 10, 15 and 20 certificates issued to it.

TABLE II
SIMULATION SCENARIOS OF VKM EVALUATED UNDER ATTACK.

| PARAMETERS | VALUE |
|---|---|
| Network dimension | 1000 x 1000 metros |
| Transmission range | 120 meters |
| Nodes | 100 |

| | |
|---|---|
| Mobility model | Random waypoint |
| Max speed | 20 m/s |
| Maximum pause time | 20 seconds |
| Simulation Time | 1500 seconds |
| Propagation Model | Two-ray ground reflection |
| Media Access Protocol | IEEE 802.11 |

## A. *Evaluation under Sybil attacks*

The Sybil attack is characterized by a malicious node creating several fake identities to itself. It can be extremely harmful to any voting based protocol, for example.

The use of certificate chains makes the PGP-Like highly vulnerable to this attack, as shown in Figure 4 and 5 [18]. The percentage of nodes with false identities in their local repositories is extremely high, reaching 80% after 800 seconds of network lifetime, independently of the number of attackers, 5%, 10% or 20%.

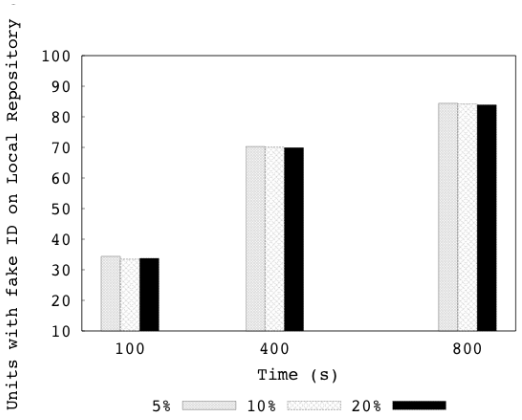


Figure 4. Trust in Fake Identities in PGP-Like.

Figures 4 and 5 also demonstrate the number of authenticated false identities by correct nodes. Note that an attacker $N$ might create a false identity $f$ and issue fake certificates to $f$. All nodes which trust $N$ will also trust $f$. Therefore, if the attacker $N$ has a correct behavior for a considerable time, several units are likely to trust it, as the false identity is spread throughout the network due to the certificate exchange mechanism.

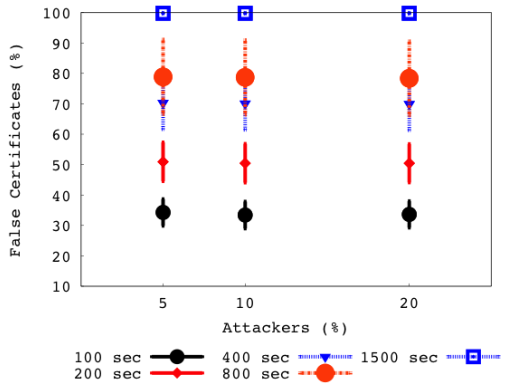


Figure 5. False Certificates in Local Repositories in PGP-Like.

In GKM, a false identity must build a group to enter the system. Moreover, the group must contain at least two nodes from other groups, i.e. the group with the fake id must have at least two non-malicious nodes. Considering that the malicious node is able to build such a group, it still has to authenticate the public key of the fake id. It is necessary to find two disjoint group certificate chains to authenticate it. This is only possible if several malicious nodes participate in the system.

Figure 6 presents the impact of a Sybil attack over GKM. In scenarios with 5% or 10% of Sybil nodes, no false identity is authenticated. Further, with 20% and group size 9 (m = 9), less than 5% of false identities are authenticated, with group size of 12 no fake id authentication is performed. On the other hand, 100% of false identities can be authenticated with 40% of Sybil nodes and groups with 6 members. These results demonstrate that with less than 40% of attackers, GKM is highly resistant to Sybil attacks. It also demonstrates that the resistance of GKM depends on group sizes, larger groups provide better resistance against Sybil attacks.

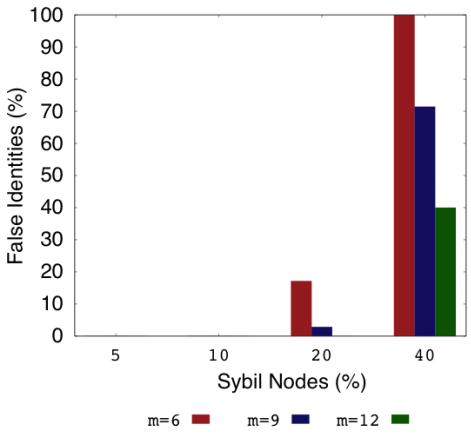


Figure 6. Authenticated false identities in GKM.

In VKM, all authentications are performed following the virtual structure. All fake indentities created by the Sybil attacker cannot be included in the virtual structure. Thus, the attack is completely meaningless, since this false identity cannot be authenticated, as they are not part of the virtual structure.

As shown in Figure 7, there is no fake identity inside the virtual structure, independently of the number of attacker and the number of certificates issued by the nodes. Consequently, no fake identity can be authenticated in VKM, making it completely secure against Sybil attacks.

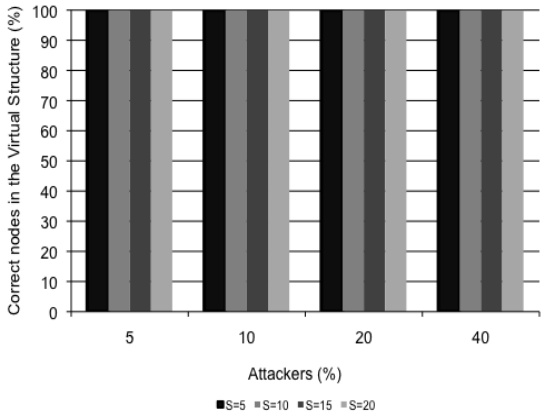


Figure 7. Correct nodes in the virtual structure in VKM.

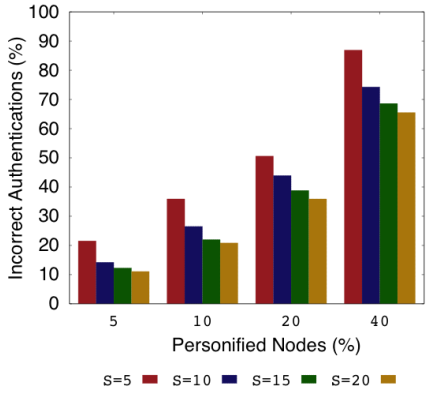


Figure 8. Authenticated false identities in VKM.

In fact, VKM is 100% secure against Sybil attacks. It implies in preloading the virtual structure in each node, but the use of the virtual structure eliminates the effect of a Sybil attack, independently of the number of attackers.

## B. *Evaluation under Personification attacks*

In the personification attack, the attacker takes the place of a leaving node and behaves as it was the correct one, or the attacker invades a valid node turning it into a malicious one. The results for the personification attack on the PGP-Like and the GKM are the same as the ones presented for the Sybil attack, in Figures 4, 5 and 6. The personification attack and the Sybil one have the same effects on PGP-Like and GKM.

On the other hand, in VKM the personification attacker becomes a part of the virtual structure. Thus, it can authenticate and be authenticated by correct nodes. Figure 8 shows the impact of the personification attack on VKM.

As shown in Figure 8, even with 20% of attackers in the network, VKM is able to correctly authenticate more than 40% of the certificates. In the presence of 5% of attackers, the VKM is able to correctly authenticate approximately 80% of certificate chains.

PGP-Like is completely vulnerable, even with only 5\% of attackers (Figure 5), while GKM becomes vulnerable with 40% of attackers (Figure 6).

Both in GKM and VKM, it is possible to see that increasing the connectivity of the virtual structure, it is still possible to reduce the effects of personification attacks. In GKM, the worst-case scenarios are the ones with the smallest groups, as it is easier for a Sybil node to join different groups. In VKM, the worst scenarios are the ones with the smallest connectivity of the virtual structure, as there are fewer virtual paths between pairs of nodes. Moreover, it is possible to increase the resistance of VKM by requesting that the source finds two distinct paths in the virtual structure for each authentication, though this improvement is considered a future work.

## C. *Scalability and Communication Overhead*

This section contains a comparison between PGP-Like, GKM and VKM considering their characteristics, scalability and communication overhead. Table III depicts all characteristics of the three schemes.

In PGP-Like, nodes can enter or leave the system at any time. In GKM, nodes can also join and leave the network at any time. When a new node wants to join the system, it must find at least $M-1$ other trusted nodes and form a group. When a node leaves the system, it does not directly affect GKM, though it is necessary that at least a predetermined number of members of the group remains in the system. If the number of members in a group is smaller than this value, it is considered excluded from the network and the remaining nodes must build another one and reenter the network.

TABLE III
MAIN CHARACTERISTICS OF PGP-LIKE, GKM AND VKM.

| | PGP-Like | GKM | VKM |
|---|---|---|---|
| Authentications | Certificate Chains | Groups Certificate Chains | Certificate Chains following the virtual structure |
| Virtual Network | -- | Dynamic according to group creation and removal | Fixed |
| Overhead | Certificate Exchange mechanism + Reactive Validation of expired certificates | Group Certificate Exchange Mechanism + Reactive Validation of expired certificates | Reactive validation of certificates |
| Scalability | Nodes can easily enter the system | Depend on new nodes forming groups | Depend on reshuffling the virtual structure |
| Local Repository Size (Memory overhead) | All nodes certificates | All groups certificates | 2*S (number of certificates issued by each node) |
| Reactive certificate validation | Yes | Yes | Yes |
| Resistance against Sybil | Completely vulnerable | Depend on group size and | Always 100% |

| | | | |
|---|---|---|---|
| attacks | | number of attackers – might be completely vulnerable | |
| Resistance against Personification Attacks | Completely vulnerable | Depend on group size and number of attackers - might be completely vulnerable | Depend on the number of certificates issued by each node and on the number of attackers – might reach 80% of vulnerability |
| All users reachability | After convergence period | After convergence period | Since network formation |

In VKM, the virtual structure is static and well known by all nodes. Thus, this scheme does not easily allow new nodes in the system, after the initialization. This characteristic is not desirable for some applications of MANETs, in which the join of new nodes is very common. However, other situations can be more predictable, as meeting communications, in which the parties are well known at the beginning of the meeting. Note that VKM allows nodes to leave the system at any time, affecting only the amount of valid certificate chains in the virtual structure.

In terms of communication overhead, all schemes are efficient. The certificate exchange mechanism of the PGP-Like and the group certificate exchange mechanism of GKM are only locally performed. In other words, nodes exchange certificates only with their physical neighbors. In VKM there is no certificate exchange, thus its overhead is zero.

The key authentication overhead of PGP-Like depends on which phase the source finds the certificate chain. If it finds in its local updated repository, the overhead is zero. It must merge its repository with the destination one and it must obtain the destination updated repository through the network. If it uses the non-updated repository, it must reactively validate all expired certificates.

GKM has the same overhead as PGP-Like, the only difference is that its overhead is over group certificates. However, it has higher communication overhead to maintain the group and to perform key update and revocation.

In VKM, all key authentications are reactive and all certificates from the chain must be validated through the network. This makes VKM the scheme with the smallest communication and memory overhead.

## V. Conclusion

Key management is a critical service in wireless ad hoc networks. It must deal with all security issues in a self-organized and decentralized way while considering nodes mobility and dynamic topology.

According to several authors, the main key management scheme for MANETs is the PGP-Like. However, it is highly vulnerable to personification and Sybil attacks. Its functionality is fully compromised with only 5% of attackers in the network.

Addressing the PGP-Like security issue against these attacks, it is possible to find in the literature another scheme, the GKM. GKM is a virtualization-based scheme, in which all operations are based on groups of nodes. It is resistant against personification and Sybil attacks if the number of attackers is up to 40%.

This article introduces the VKM, another virtualization-based key management scheme for MANETs, which is based on a virtual structure to define all key and certificate operations on the network. VKM is 100% secure against Sybil attacks independently of the network configuration and the number of attackers. It also provides a good protection against personification ones, being able to correctly authenticate more than 50% of the certificates with 20% of attackers in the network.

The article also presents a comparison between these three schemes considering the scalability and overhead issues. Even though VKM does not scale easily, it has the smallest communication and memory overhead. Future work includes the evaluation of VKM with different virtual structures and different types of attacks.

## References


[1] Johnson DB, Maltz DA. “Dynamic source routing in ad hoc wireless networks”. *Mobile Computing*, 1996; 153–181.

[2] Perkins CE, Royer EM. “Ad-hoc on-demand distance vector routing”. *Proceedings of the 2nd IEEE Workshop on In Mobile Computing Systems and Applications*, 1999; 90–100.

[3] Argyroudis P, O’Mahony D. “Secure Routing for Mobile Ad hoc Networks”. *IEEE Communications Surveys and Tutorials*, 2005; 7(3):2–21.

[4] Lundberg J. “Routing security in ad hoc networks”. *Technical Report Tik110*. 501, Helsinki University of Technology, 2000.

[5] Wu B, Chen J, Wu J, Cardei M. “A survey on attacks and countermeasures in mobile ad hoc networks”. *Wireless Network Security*, 2007; pp. 103-135.

[6] Diffie W, Hellman ME. “New directions in cryptography”. *IEEE Transactions on Information Theory*, 1976; IT-22(6):644–654.

[7] van der Merwe J, Dawoud D, McDonald S. “A survey on peer-to-peer key management for mobile ad hoc networks”. *ACM Computing Survey*, 2007; 39(1):1.

[8] Djenouri D, Khelladi L, Badache N. “A survey of security issues in mobile ad hoc and sensor networks”. *IEEE Surveys and Tutorials*, 2005; 7(4):2–28.

[9] Câpkun S, Buttyán L, Hubaux JP. “Self-organized public-key management for mobile ad hoc networks”. *IEEE Transactions on Mobile Computing*, 2003; 2(1):52–64,

[10] Câpkun S, Hubaux JP, Buttyán L. “Mobility helps peer-to-peer security”. *IEEE Transactions on Mobile Computing*, 2006; 5(1):43–51.

[11] Zimmermann PR. *The official PGP user’s guide*. MIT Press: Cambridge, MA, USA, 1995.

[12] Nogueira M, Pujolle G, da Silva E, dos Santos A, Albini LCP. “Survivable keying for wireless ad hoc networks”. *Proceedings of the 2009 IFIP/IEEE International Symposium on Integrated Network Management (IM '09)*, 2009; 606–613.

[13] e Silva RF, Silva E, Albini LCP. “Resisting impersonation attacks in chaining-based public-key management on manets: the virtual public-key management”. *Proceedings of the International Conference on Security and Cryptography (SECRYPT 2009)*, 2009; 155– 158.

[14] Christianson B. “Why isn’t trust transitive”. Proc. of the International Workshop on Security Protocols (WSP1996), 1996; 171–176.

[15] Albini L, Caruso A, Chessa S, Maestrini P. “Reliable routing in wireless ad hoc networks: The virtual routing protocol”. *Journal of Network and Systems Management*, 2006; 14(3):335–358.
[16] Robba A, Maestrini P. “Routing in mobile ad- hoc networks: The virtual distance vector protocol”. *Proc. of the IEEE Inter. Conference on Mobile Ad-hoc and Sensor Systems (MASS 2007)*, 2007; 1–9.
[17] NS-2. *The network simulator - ns-2*, 2007. URL http://www.isi.edu/nsnam/ns.
[18] Silva E, dos Santos AL, Albini LCP, Lima MN. “Quantify misbehavior attacks against the self-organized public key management on manets”. *Proceedings of the International Conference on Security and Cryptography (SECRYPT 2008)*, 2008; 128–135.
[19] Khalil M. “Improve Quality of Service and Secure Communication in Mobile Adhoc Networks (Manets) Through Group Key Management”. *International Review of Basic and Applied Sciences*, 2013; 1(3).
[20] Gharib M, Minaei M, Golkari M, Movaghar A. “Expert key selection impact on the MANETs' performance using probabilistic key management algorithm”. Proc. of the 6th International Conference on Security of Information and Networks (SIN '13), 2013; 347-351.
[21] da Silva E, Albini LCP, “Towards a fully self-organized identity-based key management system for MANETs”. Proc. of the IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2013; pp.717,723.