A Trust-Based Security Architecture for Small and Medium-Sized Mobile Ad Hoc Networks

Leonardo Augusto Martucci, Christiane Marie Schweitzer, Yeda Regina Venturini,
Tereza Cristina Melo de Brito Carvalho and Wilson Vicente Ruggiero

Abstract— **This paper describes a trust based security architecture for small/medium-sized mobile ad hoc networks. We designed and implemented a security architecture that extends the traditional PKI model, assigning variable trust values to digital certificates and issuing credentials to grant access to network services. Trust values are not static; they vary during regular network operation as network users provoke security incidents. Depending on the seriousness of the incidents the trust value associated to the offender's certificate will vary. Eventually, a series of security incidents may end up with the certificate revocation. We also developed a security framework for designing secure applications and built prototypes to test and validate our architecture. We considered service-oriented ad hoc networks, where every mobile device is classified as service providers or service users.**

Index Terms— **ad hoc networks, security, trust.**

I. Introduction

MOBILE ad hoc networks are notorious for their unusual characteristics, such as the lack of a permanent infrastructure, the sporadic nature of connectivity, the dynamically changing topology and the absence of network frontiers and central entities [8]. Mobile ad hoc networks, due to their singular attributes, demand new protocols and solutions for their open issues, such as suitable routing protocols, convenient QoS designs, applicable network addressing schemes and appropriate security mechanisms, for instance.

Security in mobile ad hoc networks is a matter of scope and environment as its requirements basically depend on the network purpose and on the network goal. For instance, the security requirements of a military ad hoc network differ according to the faced scenario. Confidentiality and availability are the most important issues in a battlefield, whereas in a humanitarian rescue mission scenario, availability is far more meaningful than confidentiality. Therefore, the application context defines the security requirements in every case.

This paper presents a security architecture designed for small and medium-sized service-based ad hoc networks whose mobile and fixed devices can be grouped under a same administrative authority. Security is achieved extending the existing PKI (Public Key Infrastructure) model. Non-static trust information was added to digital certificates and new PKI states were defined. Certificate-based authentication procedure is preceded by a group authentication technique, which works with pre-shared keys and symmetric ciphers, verifies if the devices belong to a trusted group. The group authentication is a challenge-response mechanism presented in [10].

An object-oriented application framework implements the trust-based security architecture functionalities. It was built for designing and developing application for mobile ad hoc networks over a secure foundation provided by the proposed security architecture. This framework is briefly described in this paper.

Two prototype applications (an electronic file signer and a secure slideshow application) were designed and implemented over the application framework in order to test and evaluate the security provided by the architecture. A second, but not least important, reason was to test and evaluate the usability of the framework.

The remaining of this paper is organized as follows: security threats against general ad hoc networks are briefly addressed in section II; in section III, an overview of the state of art of context-based security for ad hoc networks is provided; in section IV, the scope of the proposed security architecture and appropriate environments are described; in section V, the proposed

Manuscript received May 21, 2004. This work was supported in part by FDTE (Foundation for the Engineering Technological Development) - São Paulo - Brazil and Ericsson Research at Kista - Sweden.

Leonardo A. Martucci was with Laboratory of Computer Architecture and Networks, Department of Computer and Digital Systems Engineering, Escola Politécnica, Universidade de Sao Paulo, São Paulo - SP, Brazil. He is now with Karlstads Universitet - Institution för Informationsteknologi - Datavetenskap, 651-88 Karlstad, Sweden (phone: ++46-54-7001935; e-mail: leonardo.martucci@kau.se).

Yeda R. Venturini, Christiane M. Schweitzer, Tereza C. M. B. Carvalho and Wilson V. Ruggiero are with Laboratory of Computer Architecture and Networks, Department of Computer and Digital Systems Engineering, Escola Politécnica, Universidade de Sao Paulo, São Paulo - SP, Brazil (email: {yeda, chrism, carvalho, wilson}@larc.usp.br).

security architecture is presented; in section VI, a step-by-step roadmap on how to secure an ad hoc network with the proposed trust-based security architecture is provided; security mechanisms used to secure an ad hoc network running over the secure application framework are presented in section VII whereas implementation details, tests and results can be found in section VIII and IX; section X summarizes the achieved results and also provides a glance of future research activities.

This paper summarizes one of the results of a two-year research project held at University of São Paulo (USP) and corresponds to the third paper to be published regarding the achievements of this project. The first two publications, [10] and [16], described a security model for ad hoc network and a challenge-response mechanism used to identify trusted devices in an ad hoc environment. A fourth paper describing a more refined challenge-response authentication mechanism for ad hoc networks is going to be published in the near future [11].

II. Security Threats in Ad Hoc Networks

Network services available anytime and anywhere and wandering nodes with seamless connectivity are two important ad hoc networks characteristics. However, this absolute lack of boundaries is the Achilles heel of such networks, as no network borders exist to be defended, turning security into a fuzzy task. Therefore, every device has to guarantee its own security [7].

Security threats in ad hoc networks are somewhat an extension of the threats found on conventional (wired) networks. Even though these threats are described in several published works, such as [7], [8] and [13], we intend to provide a brief description of security threats and their relation with ad hoc network characteristics, in order to deliver enough background for the good understanding of the rest of this paper.

The security taxonomy described in [15] is used to allow the identification of new attacks that concern wireless networks.

A. Passive Attacks

Mobile ad hoc networks are passive to eavesdropping (as any wireless network), due to the communication medium nature. Interception of radio frequency carriers and, therefore, the transmitted data (that shall or shall not be ciphered), must be understood as unavoidable. IEEE 802.11 and Bluetooth, two of the most popular wireless communication standards nowadays, rely on spread spectrum (SS) communication with public direct sequence (DS-SS) codes and/or public frequency hopping (FH-SS) patterns, in order to provide interoperability among devices from different vendors. In these standards, SS does not aim security, but only ISM (Industrial, Scientific and Medical) conformance with spectrum band usage rules.

In fact, layer 1 security is hardly an option for open-standard communication technologies because a shared-medium is emulated in the physical layer. On the other hand, military communication systems are notorious for relying on long DS-SS codes or long FH-SS patterns in order to thwart passive attacks. This paper will not consider layer 1 security, as our proposed architecture was designed and implemented to be applied over open-standard wireless communication technologies.

Traffic analysis involves the capture of transmitted data, followed by their storage and analysis, in order to extract useful information from ciphered payloads. As previously seen in this section, wireless networks are exposed to eavesdropping. If weak ciphers schemes are used, its combination with passive attacks can lead to very insecure wireless networks – IEEE 802.11 WEP (*Wired Equivalent Protocol*) is an example of a poor security protocol (more about WEP weaknesses in [4]).

B. Active Attacks

Active attacks against mobile ad hoc networks are a superset of attacks on conventional networks (see more in [12] and [15]). These attacks can be divided in the following categories:

1) Replay attacks involve capturing, storing and retransmission of a message or sequences of messages. Replay attacks often prelude other security attacks. Wireless networks are highly susceptible to replay attacks, as messages are transmitted “over-the-air” and are, thus, susceptible to be intercepted.
2) Masquerade or impersonation attacks occur when one entity pretends to be a different entity. Unprotected or weak authentication mechanisms usually lead to this security threat, as message sequences can be easily replayed. Man-in-the-middle (MitM) attacks often prelude impersonation attacks. Flaws in tunneled authentication mechanisms for wireless networks using man-in-the-middle attacks were published in [3].
3) Message modification attack takes place when a message or a sequence of messages are captured or intercepted, altered and retransmitted. Intentional delaying and message reordering are also considered modification attacks. In order to prevent this kind of security attack, data integrity must be guaranteed.

Protection against modification attacks is essentially based on the same suite of protocols in wireless and conventional networks. However, mobile ad hoc networks are more susceptible to message modification, as data can be relayed by every node, trusted or not, in the wireless network.

4) Denial of service (DoS) prevents or inhibits service providing in computer networks. Logical DoS may be avoided if a strong authentication mechanism is applied, but physical DoS is hard to prevent in standardized communication systems for public usage. Service disruption in wireless networks is easy to perform, as it is possible to jam the frequency range being used by wireless communication (as it is standard defined). However, in order to jam a wireless network, the attacker must be in network range. Wireless network devices are also susceptible to battery exhaustion attacks, a special kind of denial of service that targets battery-driven mobile devices [14].

III. The State of Art of Context-Based Security for Ad Hoc Networks

As presented in section I, defining the context and the purpose of a mobile ad hoc network is decisive as it sets the security demands for each specific scenario. This section presents the state of art in the security for ad hoc networks, presenting security models, mechanisms and also their target scenarios. The most relevant works concerning the context and the scope of our work (see section IV for more information about the scope) are also presented in this section. Nevertheless, we do not have the intention to present an exhaustive list of published works regarding ad hoc networks security.

"Spontaneous network" proposal [7], for example, was designed to secure ad hoc networks restricted to a small area, such as a room, where users can share a common secret and set a secure and spontaneous network. A similar approach was proposed in [2], which assumes the same scenario as starting point, but a slightly different goal - setting strong symmetric keys from weak ones. As seen, both proposals were designed for a very specific environment, a small group confined in a closed place, like a meeting room or a conference room. In addition, all users must trust each other, which is a reasonable assumption for a closed context.

An alternative and realistic scenario is an environment where all devices belong to one user or a group of users or even a small office. All these devices are under a same administrative authority and they all belong to the same secure group of devices, which may establish secure communication channels among them. The setting of these groups and the distribution of cryptographic keys among devices that compose a group were the target of several papers. "The Resurrecting Duckling" security model [13] and the following "What's next?" [14] were among the first works to propose a solution for this scenario using a central and portable device, the "cyber representative", which distributes digital certificates to other devices using physical contact, in a process denoted "system imprinting". This model was the first security design that was complete enough to be denoted as security architecture for mobile ad hoc networks to be ever published. It tries to cover all network threats in a single and coherent solution. However, this proposal is far from perfection due to some naïve assumptions, such as an all-mighty device, the absence of a closed scope and the lack of proper solutions for some security questions, such as battery exhaustion attacks.

Zhou and Haas [18] presented a mechanism for key setting and distribution in an ad hoc network distributing pieces of a private key among special devices denoted servers and signing certificates using threshold cryptography. This mechanism was later improved in [9] by allowing a group of nodes that share a common secret to sign a digital certificate. Although none of these two papers specify a target environment, they are obviously meant to be applied in closed environments, where nodes know each other beforehand, as they are supposed to share some sort of common information before starting to issue certificates. Therefore, it is reasonable to assume that both of these mechanisms, even though being designed to secure routing in ad hoc networks, rely on the assumption that at least its first nodes belong to a single user or community of users that share a common interest.

Hubaux, Buttyan and Capkun [8] proposed a public-key distribution system suitable for self-organized ad hoc networks. Their proposal remembers PGP (*Pretty Good Privacy*) system, with users issuing their own digital certificates, but with no directory server for public key distribution. In fact, this work suggests that every device shall keep a small repository with chosen certificates selected by the user. Public-key checking is done by merging the local repositories of two users/devices and trying to find a certificate path (chain) between them. However, the presence of dishonest users is poorly addressed and new authentication methods are needed to circumvent this problem. This system was

designed assuming a network that exhibits a *small world* property (see [17]). The *small world* scenario, applied to the ad hoc networks environment, postulates that these networks have a small average diameter and highly clustered characteristics, which increase the probability of finding a certificate path between two nodes. They assert that their proposal can be applied in self-organized environments, such as ad hoc networks and peer-to-peer applications, but its usability seems to be very limited to users will, and it seems not suitable for automatic activities (e.g. data synchronization).

Candolin and Kari presented a security architecture for wireless ad hoc networks relying on trust information [5]. Even though no specific environment(s) is (are) explicit in the paper, some architecture details, such as a network establishment along with a certificate issuing procedure, reveal the nature of the target ad hoc scenario (small ad hoc networks that can rely on a single certificate issuer). Trust information is service-oriented, which means that a device shall have multiple trust values associated to it and decisions are based on the trust relationship between service provider and user. However, how exactly trust information is first set and also how trust loss occurs is omitted. Furthermore, the revoking method can lead to full-scale DoS attacks against the protected ad hoc network, as a compromised node can falsely declare that a network device is guilty of an offensive action, which may lead to the revocation of the victim's certificate.

In next section, the scope of our security architecture is presented and also its target environment, as well.

IV. The Scope and Environment

A trust-based security architecture suitable for small and medium-sized mobile ad hoc networks is the main goal and contribution of this paper. However, before starting to describe the security architecture and its implementation we need to define the terms small and medium-sized ad hoc networks.

We consider small and medium-sized networks all networks whose devices belong to a single person, a group of persons (e.g. a family) or an organization (e.g. an office, a small community). In fact, we believe that the great majority of future mobile ad hoc networks will fit under the given specification. In addition, we believe that some small administrative work is acceptable to perform some key actions (e.g. joining new devices to the secure network) for an ad hoc network with a limited number of users and devices.

We also considered a service-oriented network (Jini-like [1]), where all devices are classified as service providers or users. Service-oriented networks usually have one or more service directory services, which track and keep a list of all available network services in the neighborhood. We assumed that any network device with enough resources can assume the role of service directory in the absence of an online directory service. And we supposed that mobile devices could be clustered according to its ownership or affinity (i.e. personal devices from employees of one office may belong to the office's secure cluster and also to the employee's home cluster, as well). These secure clusters are named *virtual domains* [10].

We also presupposed that a secure wireless ad hoc network established under the rules of the proposed security architecture is under control of at least one person with administrative rights (administrator), as usual in any existing network. Administrator roles include: initializing and creating a new secure ad hoc network, allowing devices to join a secure network, expelling devices from a secure network, etc.

Even though all devices belonging to a single administrative authority may be scattered and out of radio range, they will still keep their bounds with other devices belonging to the same secure network. In fact, the terms small and medium-sized refer to the size of the secure network only. Moreover, this limitation regarding the dimension of the ad hoc network is given only because some administrative work is needed during system bootstrapping, as described in the following sections. Notice that a secure network can be established and run over an insecure ad hoc network.

V. Trust-Based Security Architecture

The proposed trust-based security architecture for small and medium-sized ad hoc networks assumes a service-oriented network. Network devices "incarnate" network service providers and/or clients.

The proposed architecture is a composition of shared secrets, loose synchronization among devices' real time clocks, symmetric and asymmetric ciphers protocols and trust information embedded in digital certificates. Next, some assumptions about network devices to be secured are made:

1) Every device has to run at least one symmetric and one asymmetric cipher and has memory enough to store its own digital certificates;
2) Some devices have enough memory to keep a list of

all running services in the neighborhood (defined by radio range). All these devices are eligible to host a service directory;

3) At least one device has a user-friendly interface and enough memory to keep a digital certificate store.

We stress that the main goal of the proposed security architecture is the achievement of a security architecture blueprint and an application framework as well, in order to provide a platform for implementing new secure applications in such environments.

In this section, we introduce the main components of the proposed security architecture. In subsection A we present the network entities, which a common definition for every physical or logical component of the proposed security architecture. In subsection B we classify these entities according to their status and in subsection C we introduce how trust information is spread in the ad hoc network and translated in our security architecture. Finally, in subsection D, we present the new trust states that extend the current PKI model.

Fig. 1 shows the security architecture components and some relationships between them. It represents generic entities, their contents (*status* and *trust information*) and also how trust information is changed (*reports* and *new trust*). The *trust management* block is actually a service of an existing entity, as seen in this section.

A. Network Entities

Logical entities run in physical devices. One device may host one or more logical entities. The proposed security architecture assumes two different entities:

1) Clients;
2) Services - three different service providers exist: lookup services; registration services; and general service providers.

The term entity will be freely used in this paper to refer services and also clients.

Clients (C) are entities that use services. They may either be a piece of software or a human user interacting with a mobile device. General service providers (P) are entities that deliver services. Services may be provided for public access or restricted to known entities. Peers that request services are denoted clients and peers that receive service requests are denoted service providers.

Lookup services (LS) are directories that keep a list of all available network services in the neighborhood, which is defined by radio range of the wireless interface. One or more LS may exist at the same time and any device with enough resources in the ad hoc network may run a LS in no LS is available.

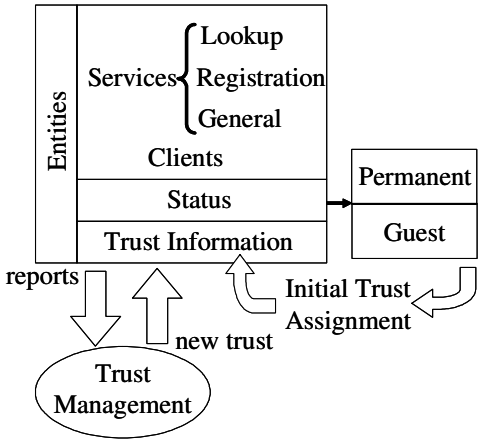

Fig. 1. Security Architecture items: entities (clients or servers), their related status and trust information.

Registration Services (RS) are the first service and starting point of every ad hoc network to be secured with our architecture proposal. A device that runs a RS needs a friendly user interface and is supposed to be a resourceful mobile device, with memory and processing power enough to keep a small digital certificate database and to issue digital certificates in a reasonable response time. RS issue digital certificates with embedded trust information and keep lists of issued certificates and modification of their trust values. RS have similarities with PKI's Certificate Authorities (CA). However, RS really extend the CA concept. For instance, RS can change clients and service providers' privileges by issuing or revoking, upon request, digital certificates that are not only meant for identification purposes, but for refining access-control. We denoted this family of certificates as credentials because they provide restrictions and grant access to network services.

In addition, RS may renew certificates and distribute and renew shared secrets among devices that belong to the secure ad hoc network (*virtual domain*).

RS also track the behavior of clients and service providers through security events, which are reports of network offenses, perpetrated by clients or service providers against their peers. Security events may also report nice and good behavior and RS translate these events in changes in trust information regarding one entity. Moreover, the starting point of a new secure ad hoc network is a RS with a self-signed digital certificate. Furthermore, a RS with a self-signed digital certificate can issue certificates to other entities join the secure network and even to other RS, which hold a certificate issued by the first RS that allows them to also issue certificates. Other entities can only join a secure network through an interaction with a RS. A device may host several services and clients.

B. Entity Status

Service providers and clients are classified according to their current entity status:

1) Anonymous guests or;
2) Identified guests or;
3) Permanent entities.

Permanent entities have long-term privileges, which are set during an initial configuration process.

Guest entities are capable of starting a communication channel and use services, but they have few privileges and rights. Identified guest have short-term rights and must be submitted to a registration process. Anonymous guests are users that were not submitted to a registration process and, hence, cannot be identified. In addition, anonymous guests can use only public services.

An initial configuration process sets the entity status and also its privileges. The initial configuration process runs only in RS. This process is manual, giving to the network owner total control over user rights. The initial configuration process is also used to register incoming devices and grant service credentials, providing a better control over the secure ad hoc network. The spread of the configuration data over the network occurs naturally, without any user intervention (details are provided on section VI).

C. Trust Information and the Network Perception

When service providers or clients join a secure ad hoc network, they receive a certificate from a RS.

A digital certificate received from a RS carries more information than a regular certificate (i.e. version, serial number, issuer, expiration date, etc.). It also has a trust value that tells the maximum trust that this entity will have whereas it bears the certificate in question.

As previously and briefly stated in subsection A, trust information is important since it works as a service access control parameter, granting or denying network rights to use services.

Moreover, trust information translates the *network perception* about one entity. *Network perception* can be understood as a network's common intelligence regarding one entity and it is determined by its behavior towards the rest of the network in terms of security. RS are responsible to translate entities behavior in new trust values and also to inform the new trust information among the network entities.

Therefore, trust regarding one entity may rise or fall according to its behavior. If a client commits a fault against a network print service for instance (e.g. printing 50,000 high-quality copies of a book - a DoS attack), its trust value may fall. Entities behavior must be reported to a RS in order to have their behavior translated in new trust values.

However, in ad hoc network environments, it is not possible to count on specific services to be available at all times, as they can be out of range or even turned off. In order to make our architecture compliant with ad hoc network characteristics without compromising security, two inner mechanisms were designed:

1) A *local perception* on every entity, which is an instant reaction mechanism used as immediate response against attacks and can deny access to local services as soon as an attack is identified;
2) In order to report faults to RS, a *gossip mechanism* is used. The *gossip mechanism* works as follows: when an entity is attacked by another entity, it first tries to report the security event to an available RS, but, if none are present, it keeps the information regarding the fault and waits a RS to be in radio range (if no ad hoc routing protocol is running). Once a RS is available in the network neighborhood, the entity sends, or gossips, as we prefer, all stored data regarding network security attacks to the RS.

To build the *network perception*, fault reports must be consolidated in order to obtain the current picture of the network trust information. However, if a secure ad hoc network has two or more RS, each RS will hold a small piece of the actual *network perception*. Merging trust information from different RS demands synchronization.

RS keeps a list of all received security reports. Before synchronization, each RS stamps its lists with a version number and its name (e.g.: RS^A). When two RS meet, they do not only exchange their own report lists, but also verify if one of them has a more recent report from other RS that it is not available at that time.

Synchronizing trust information periodically or when a considerable amount of reports is available causes an obvious delay in the *network perception* consolidation process. On the other hand, synchronization every time a new trust report is received can significantly impact the network traffic and cause a waste of battery resources from RS. However, in ad hoc network environments, it is not guaranteed that all RS of a given *domain* are available at all times. Therefore, report synchronization among RS can be delayed or occur not simultaneously among all RS (if more than two RS exist) what implies having RS with different *network perceptions* at the same time.

For instance: if three RS exist in a given domain (RS^A, RS^B and RS^C) but only two of them (RS^A and RS^B) are available during synchronization time, these two RS exchange their most recent report lists and verify if any

of them have a newer version of RS^C's list. If RS^B leaves the ad hoc network and RS^C arrives, RS^A and RS^C can synchronize their lists and, moreover, RS^C will also get an updated version of RS^B report list, as RS^A had obtained this before directly from RS^B. Notice that the *network perception* of all three RS is not the same at any moment in this example. In fact, in real ad hoc networks, *network perception* will hardly be the same in all RS as mobile nodes can leave and join the ad hoc network at anytime. And if RS are never available at the same time, RS may demand that entities with enough memory and processing resources to store a local version of its trust report table, in order to increase the probability of this report list to reach another RS. Meanwhile, RS will carry its own *network perception*. This mechanism is only turned on by one RS if it considers that trust reports from other RS are outdated.

This natural latency in the propagation of trust reports is the trade-off between having an instantaneous picture of the *network perception* and mobility in the trust based security architecture proposed in this paper. However, if the achievement of a unique network perception is hard to obtain, *local perception* is used to thwart attacks, in order to protect a device even if the available network perception is not up to date.

D. Trust Information and Certificate Revocation List

Trust information changes are published by RS using the Trust Information and Certificate Revocation List (TICRL), which is an extension of a regular Certificate Revocation List (CRL) from PKI. Besides adding trust information, TICRL also supports three more additional states besides the CRL revoked state. TICRL lists:

1) *Active entities* are all entities that had any change in its trust information.
2) *Suspended entities* are entities that had a sudden loss of trust in a short period of time and, therefore had all their rights suspended for a determined period of time.
3) *Blocked entities* are entities whose rights were suspended for an undetermined period of time. Only the network owner or one user with administrative rights can unblock an entity.
4) *Revoked entities* are entities whose certificates were revoked. The *network perception* regarding any entity whose certificate is revoked is of full distrust.

This new form of handle certificate status establishes an extended model for digital certificates life cycle. Fig. 2 presents this life cycle.

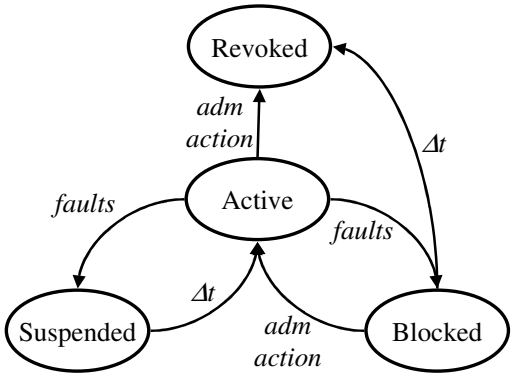

Fig. 2. Life cycle of a digital certificate in the proposed security model.

VI. Roadmap To Secure Ad Hoc Networks

In this section we present a step-by-step roadmap on how to secure an ad hoc network with the proposed trust-based security architecture. In subsection A we show how a secure network starts from a RS. Subsection B presents how a client uses a service and how a report is sent to a RS. Finally, in subsection C, we show how the TICRL are updated.

A. Step by Step: Building a Secure Ad Hoc Network

To start a secure ad hoc network, human interference is needed. First, a suitable device with a friendly-user interface must be selected to run a RS by the human owner of the network. After the RS application has been started, this primal RS auto-signs its digital certificate, thus creating a new secure ad hoc network, or *domain*, and produces a long random number that will be used to secretly identify all network entities that belong to its *domain*. After that, the network owner pre-registers in the RS all devices that he/she wants to belong to the secure network (e.g. notebooks, palm devices, etc.). All devices that join a *domain* have an entity status, alias and initial authentication method (which might be a biometric scheme, a weak password or both). The network owner may also add new RS to the *domain*. This phase is called *initialization phase*.

When an entity requests to join the domain, the RS asks for the tuple "*alias, authentication data*", and if it is correctly provided, the RS signs the device's public certificate and sends it along with the random number that identifies the domain back to the requesting entity. This phase is called *joining phase*.

For devices with no user input interface, a Bluetooth like approach is recommended (stamping a random factory short-length code in the device chassis for initial authentication purposes).

The *initialization* phase is the only operational phase that requires manual intervention or administrative

work. In fact, the need of a manual system bootstrapping is the reason of limiting the scope of this security architecture to small and medium sized mobile ad hoc networks, as it is clear that during regular operation the proposed architecture is also suitable to large mobile ad hoc networks.

The public certificate and the random number are both ciphered before being transmitted. The symmetric cipher key is derived from the authentication protocol.

The digital certificate distribution method can also be made using a side-channel distribution, as presented in [6] and [13]. This method requires both devices to be at zero-hop distance. The proposed security architecture applies a simple approach that can be executed at any distance, ciphering the public key of the requesting entity using data derived from the authentication protocol as symmetric key (see more on section VII).

The initial trust value is defined according to the initial authentication method and the entity status. A permanent entity receives a greater trust value if it was initially authenticated using a biometric method plus a password than another one using only a password as authentication method, for instance. And guest entities always receive a lesser trust value than a permanent entity. A trust-based certificate is shown in Fig. 3.

Notice that trust information is a composition of three complementary percentages: trust, distrust and unknown factor. Trust and distrust definitions are straightforward. Unknown factor represents the lack of previous behavior knowledge about a single entity.

B. Step by Step: Using a Network Service

After issuing trust-based digital certificates to entities, the network is now able to start offering services over a secure application framework. When a Client (C) wants to use a service, it starts a secure communication with Lookup Service (LS) and asks for a Service Provider (P) that provides a given network service (e.g. printing).

LS keeps a list of all P available in the domain and verifies if the requested P (e.g. P_1) is currently available. If P_1 is available, LS sends P_1 address to the requesting client (C_1). Client C_1 tries to establish a secure communication with service provider P_1, which verifies if C_1 has enough trust to use the requested service and if it also has all the needed credentials (if any is needed).

P_1 may also look for an available Registration Service (RS) to get the current *network perception* about C_1. The same procedure is followed by C_1 regarding P_1 status. If C_1 and P_1 requirements are both fulfilled, a secure channel is established between them and the service is provided, otherwise communication ends.

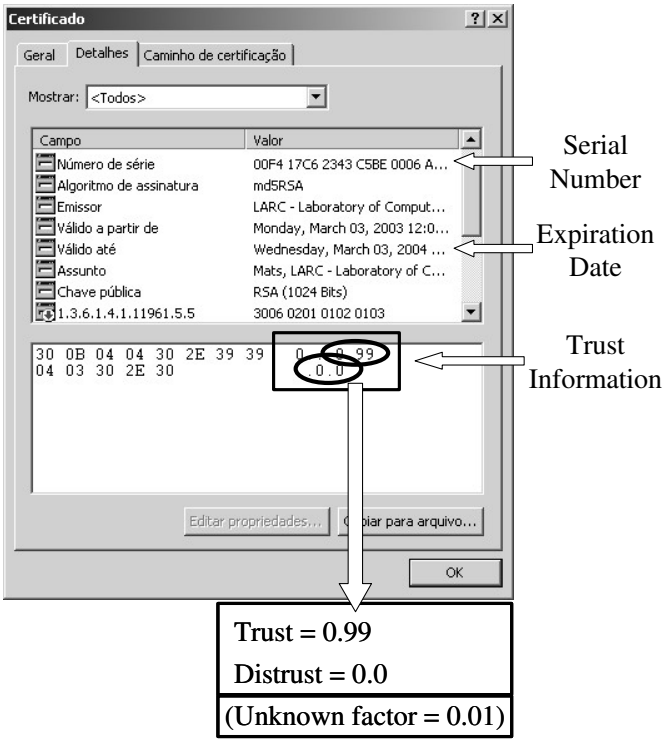

Fig. 3. A digital certificate with embedded trust information (initial trust value of 0.99).

If a security attack is detected by either C_1 or P_1, a security report regarding the offender is generated by the offended entity. The offended entity queries LS for an available RS. If an RS is available at that moment, the security fault is reported; otherwise it is stored and kept until a RS becomes available.

Security events are classified in six categories (we understand that six levels of security events offer at the same time a good granularity and simplicity):

- Three regarding network offenses (incidents), from critical to light offenses;
- Three regarding nice network behavior (e.g. absence of security faults in a given period of time, extreme security awareness, etc.).

As definition of a security event may change from entity to entity, they are full responsible for classifying network offenses and delights according to the proposed six-level classification.

C. Step by Step: Updating Trust Tables

Once a security event is reported to a Registration Service (RS), it add it to its Trust Event List (TEL), which contains the history of all reported events, and calculates the new trust value for a given device.

If multiple RS exist, each RS builds one TICRL regarding only certificates issued by it. Therefore, if RS^A issued certificates for entities C^{A1}, P^{A1} and P^{A2}, it will only build a TICRL regarding these three entities.

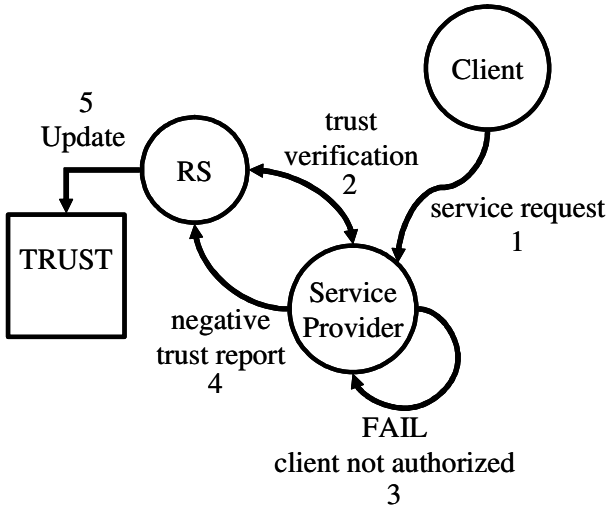

Fig. 4. A trust update is represented here. First, a Client requests a service to a Service Provider (1). The Service Provider verifies if Client has enough rights for the requested service consulting a locally stored copy of TICRL or requesting it to an available RS (2). If the trust associated to the Client is not enough (3), a negative trust report is sent to an available RS (4). The RS updates its report list and the TICRL.

However, RS^A may have security events regarding entities with certificates issued by RS^B in its TEL. RS^A and RS^B have to synchronize their TEL in order to build a unique TICRL that can offer a true picture of the current *network perception*.

Fig. 4 shows the workflow of trust updating through an example. It starts with a Client requesting a service to a Service Provider, which verifies the Client access rights, and sends a trust report to the RS. The workflow ends with the update of the TICRL in the RS. Fig. 5 illustrates a TICRL of a single *domain* with three RS (RS^A, RS^B and RS^C).

VII. Security Mechanisms

The security mechanisms, used to secure an ad hoc network running over the application framework of the proposed trust-based security architecture for small and medium sized ad hoc networks, are:

1) Shared-secret network authentication followed by the establishment of a TLS secure channel and;
2) Access-control based on trust information.

The shared-secret network authentication uses the mechanism described in [10], and it aims to recognize if the communicating parties belong to the same *domain*. Furthermore, this shared-secret network authentication can attenuate battery exhaustion attack attempts, as it is based on a lightweight protocol and occurs before any high power-demanding algorithm, as asymmetric key ciphers.

RS#	Certificate#	State	Trust
RS^A	C^{A1}	**Active**	**(%,%)**
	C^{A2}	**Suspended**	**(%,%)**
	P^{A4}	**Active**	**(%,%)**
	P^{A7}	**Active**	**(%,%)**
RS^B	C^{B2}	**Blocked**	**(%,%)**
	P^{B3}	**Revoked**	**-**
	SL^{B1}	**Blocked**	**(%,%)**
RS^C	C^{C1}	**Active**	**(%,%)**
	C^{C2}	**Suspended**	**(%,%)**
	C^{P3}	**Revoked**	**-**

Fig. 5. TICRL translates the network perception of a domain with three RS.

The network authentication sets a secure tunnel between two entities (that only know that the other communicating party belongs to a known *domain*).

Inside this secure tunnel, a TLS authentication is started, with the certificates traveling ciphered inside the established channel. Therefore, each party can identify its peer univocally, but their identities travel protected from eavesdroppers inside the secure tunnel. In addition, a TLS tunnel is established between the peers and the original channel set using the network authentication mechanism is then abandoned. The service is then requested and provided inside a secure TLS tunnel.

VIII. Application Framework

The application framework is a software infrastructure designed to provide a platform for implementing new applications over a secure environment.

It was fully designed in Java in order to be platform independent. The application framework implements the security architecture and its components. It also provides an application program interface (API) for designing network clients (C) and service providers (P) over a secure infrastructure. Fig. 6 illustrates the application framework layers. A brief description of the framework layers and its functionalities is presented next:

1) A Communication Layer that is used to set a TCP connection between mobile devices.
2) A Security Mechanism Layer, composed by two sub layers. A network authentication sub layer, which verifies if a communicating party belongs to a known *domain*; and a TLS layer used to exchange digital certificates and establish a secure tunnel.
3) A Trust Layer that verifies the trust information

regarding a digital certificate. It calculates new trust values from network events (when needed) and also queries the RS for the current *network perception*.

4) Application Support Layer that has infrastructure for basic network services (RS and LS) to run and also for network clients and service providers' design. The application program interface (API) for the developing of new applications is over the client and service provider sub layer.

IX. Application Prototypes

Two prototypes were designed to test our application framework usability:

1) A *digital signer* of electronic files. It is composed by a client that requests files to be signed; a signer that receives files, evaluate the *network perception* of clients, check credentials and enable files to be signed. Signing is only done after approval of the signer owner and a verifier that checks the signature authenticity.
2) A *secure slideshow* that multicasts slides to entities that belong to the same secure ad hoc network. It was designed to be used for education support, in classrooms or meetings rooms.

The implementation of the digital signer was done just after the application framework was developed. It was designed and programmed by the same developing team as the first test of the application framework.

In this prototype, were considered faulty behaviors actions like: clients sending virus infected documents to the digital signer application and non-authorized service requests (i.e. not enough trust).

A single programmer developed the second prototype in a two-month period and with almost no assistance. This prototype was built to evaluate the usability of the application framework. Results were encouraging for a two-month period, as the developer had very little experience on Java programming at that date.

In this prototype, insufficient rights to request access to the secure slideshow content were faulty behaviors. From the client point of view, a secure slideshow server could present a fault behavior if the announced content did not correspond to the real broadcasted content. In this case, faults were not automatic detected, and user intervention was needed.

Regarding security, both applications performed well. We forced one client to commit several faults, and then checked the client's *network perception*. Reports were sent from services to RS through the *gossip mechanism*.

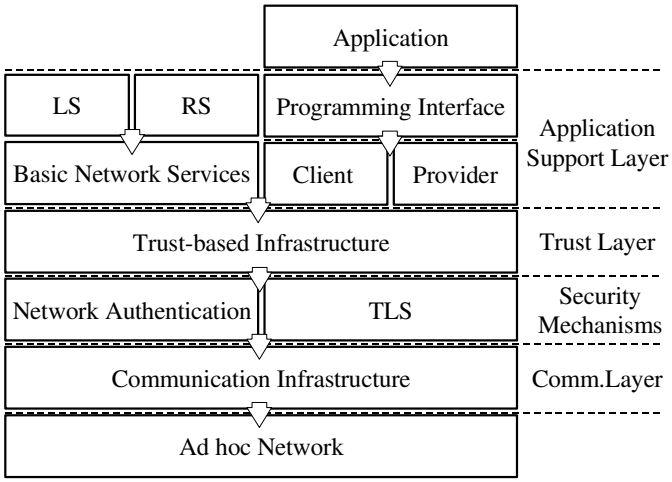

Fig. 6. The application framework and its several layers.

The fluctuation of the trust information of this user is presenter in Fig. 7. The initial trust value assigned to this device was 0.8 for trust level, 0.15 for unknown factor and 0.05 for distrust level.

After six network offenses (incidents), the trust level and the distrust level are approximately 0.5 each. And after ten incidents the trust level was 0.25 and the distrust level was of 0.75. The unknown factor naturally tends to zero, as more information is obtained about the entity's behavior in the secure network.

In this example, we artificially suppressed the suspended and blocked states of the RS, as it would first suspend and then block offender network user before the trust information reaches such low levels, as 0.16, after twelve incidents (or before the distrust level reaches 0.84, after the same twelve network offenses).

X. Conclusion And Summary

In this paper we introduced and described a trust-based security architecture for small and medium sized mobile ad hoc networks. Albeit we have limited the scope of our proposal to small and medium sized mobile ad hoc networks, the proposed security architecture can clearly be applied to larger ad hoc networks with several active RS during operation mode. We have limited our scope mainly because system bootstrapping is a manual activity. Therefore, for large ad hoc networks, an initial configuration effort equivalent to the ad hoc network size is needed.

The proposed security architecture is also suitable for ad hoc network characteristics, such as mobility, lack of network borders, dynamic topology changing, etc. (see more in section I).

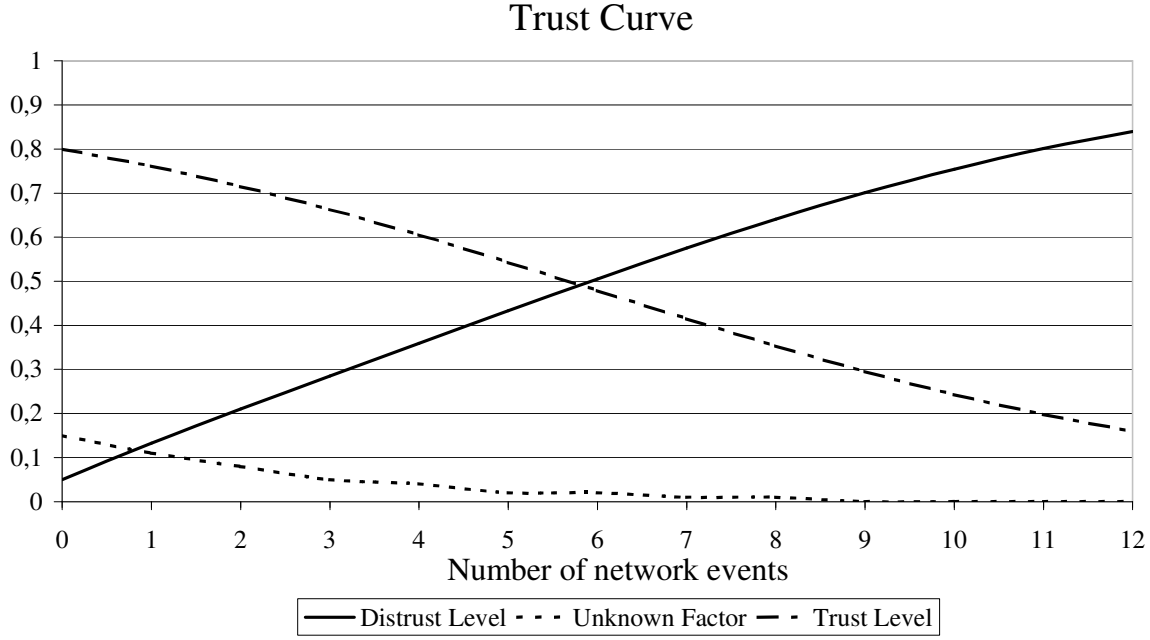

Fig. 7 Example of the trust information fluctuation of an offender network user.

Node mobility mainly impact report synchronization among RS, but, as shown, regular entities can be used to propagate report lists among RS. Even though latency exists in the consolidation of the *network perception*, a protection mechanism, the *local perception* can be used to protect entities under attack. Other effects of mobility and dynamic topology do not affect security, but only regular usage of network services (e.g. if a client looks for a non-available service, no service can be provided). Regarding the lack of network borders, virtual borders are defined using *domains* as a first stronghold to protect entities against attackers.

Summarizing, first we surveyed the security threats in ad hoc networks, classifying them according to the security taxonomy presented in [15] and focusing on the aspects regarding wireless networks. In addition, we provided the state of art of context based ad hoc networks, listing the most relevant papers in ad hoc network security field, concerning the scope of this work, and their application context.

Furthermore, we have presented and described a trust-based security architecture for small and medium-sized ad hoc networks, which assumes a service-oriented, Jini-like, network environment. We have assumed four basic kind network entities: clients, specific service providers, directory or lookup services and registration services, which extend the certification authority (CA) concept from PKI, as trust information and credentials are added to service access-control. Every entity must belong to one or more secure ad hoc networks (e.g. home network and/or office network, for instance), which is denoted *domain*.

In addition, RS track entities behavior, through a *gossip mechanism*, where entities report secure events (offenses and also nice network behavior) regarding other network entities. The RS then analyze all received security events and reduce or restore entities trust values. Trust information is published in TICRL and it reflects the *network perception*. In fact, TICRL extends the PKI model, with new states besides the *revoked*, they are: *active*, but with trust loss; *blocked*; and *suspended*. We have also shown how the trust information lists are synchronized among several RS in ad hoc environments.

In fact, mobility related characteristics, like leave and join operations, affect the proposed security architecture in trust synchronization only, as it is not possible to guarantee that the *network perception* of all existing RS is the same at all times. Other network entities (LS, P and C) are immune to mobility related characteristics in terms of security. The only effect over those entities is that they will not be able to report faults or nice behavior to the RS. Other issues non-related to security are common to any ad hoc environment, such as a client not finding a specific service in the ad hoc network.

The proposed security architecture relies on standard authentication and cryptographic algorithms, such as TLS, and non-standard security mechanisms, such as group authentication (see [10]). We have also briefly described the application framework that was designed after the proposed security mechanism, and also two prototypes that were built over this framework.

Finally, as we have observed running the prototypes,

the proposed security architecture prevents active attacks against mobile ad hoc networks. We believe the great majority of future ad hoc networks will be of small and medium-sized networks, what makes our solution a very comprehensive one, but we have also shown that the limitation of scope is only due to the manual system bootstrapping needed during *initialization* phase.

FUTURE WORK

In the near future we intend to provide a detailed performance evaluation of the trust-based security architecture, including its operational costs and timings. Furthermore, we will also provide other project results, such as results regarding power consumption gains obtained group authentication in hostile environments and others regarding cryptographic performance of the cipher sets applied and also about trust management.

ACKNOWLEDGMENT

We are thankful to our colleagues Armin Mittelsdorf, Fernando Redigolo, Cesar Rossi, Fabio Taroda and Rony Sakuragui for their invaluable contributions to this work. We also thank Mats Näslund and András Mehes for useful comments and discussions and Frank Bodinaud for testing the application framework.

REFERENCES

[1] K. Arnold. et al., *The JiniTM Specification*, Reading, MA, USA: Addison Wesley, 1999. 385p. 1999.

[2] N. Asokan and P. Ginzboorg, "Key agreement in ad hoc networks. Computer Communications", v.23, i.17, pp. 1627-1637, Nov. 2000.

[3] N. Asokan, V. Niemi and K. Nyberg, "Man-in-the-middle in tunneled authentication protocols". Technical report 2002/163, IACR ePrint archive, October 2002. Available at: <http://eprint.iacr.org/ 2002/163/>.

[4] N. Borisov, I. Goldberg and D. Wagner, "Intercepting mobile communications: the insecurity of 802.11", in *Proc. 7th Annual International Conference on Mobile Computing and Networking*, New York, NY, USA: ACM Press, 2001, pp. 180-189.

[5] C. Candolin and H. H. Kari, "A security architecture for wireless ad hoc networks", in *Proc. Military Communications Conference* - MILCOM 2002, Piscataway, NJ, USA: IEEE Computer Society Press, 2002, v.2, pp. 1095-1100.

[6] S. Capkun, J. P. Hubaux and L. Buttyan, "Mobility Helps Security in Ad Hoc Networks", in *Proc. 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, New York, NY, USA: ACM Press, 2003, pp. 46-56.

[7] L. M. Feeney, B. Ahlgren and A. Westerlund, "Spontaneous network: an application oriented approach to ad hoc networking". *IEEE Communications Magazine*, vol. 39, i. 6, pp. 176-181, June 2001.

[8] J. P. Hubaux, L. Buttyan and S. Capkun, "The quest for security in mobile ad hoc networks", in *Proc. 2nd ACM International Symposium on Mobile Ad Hoc Networking and Computing*, New York, NY, USA: ACM Press, 2001, pp. .

[9] H. Luo, et al., "Self-securing ad hoc wireless networks", in *Proc. 7th IEEE Symposium on Computers and Communications*, New York, NY, USA: IEEE Computer Society Press, 2002, pp. 567-574.

[10] L. A. Martucci, T. C. M. B. Carvalho and W. V. Ruggiero, "Domínios Virtuais para Redes Móveis Ad Hoc", in *Proc. 21st Simpósio Brasileiro de Redes de Computadores*, Natal, RN, Brazil: Sociedade Brasileira de Computação, vol. 2, 203, pp. 599-614.

[11] L. A. Martucci, T. C. M. B. Carvalho and W. V. Ruggiero, "A Lightweight Distributed Group Authentication Mechanism", in *Proc. 4th International Network Conference*, Plymouth, UK, to be published.

[12] A. J. Menezes, P. C. V. Oorschot and S. A. Vanstone, *Handbook of Applied Cryptography*. Boca Raton, FL, USA: CRC Press, 1996. 816p.

[13] F. Stajano and R. Anderson. "The resurrecting duckling: security issues for ad hoc wireless networks", in Proc 3rd AT&T Software Symposium, Middletown, NJ, USA, 1999.

[14] F. Stajano, "The resurrecting duckling: what next?", in Proc. 8th International Workshop on Security Protocols, Lecture Notes in Computer Science, Springer-Verlag, Apr. 2000. Santa Barbara, CA, USA, 2000.

[15] W. Stallings, *Cryptography and network security: principles and practice* (2.edition), Upper Saddle River, NJ: Prentice Hall, 1998. 569p.

[16] Y. R. Venturini, et al., "Security model for ad hoc networks", in *Proc. International Conference on Wireless Networks*. Las Vegas, NV, USA: CSREA Press, 2002, pp. 185-191.

[17] D. J. Watts. *Small Worlds: the dynamics of networks between order and randomness*. Princeton, NJ, USA: Princeton University Press, 1999. 266p.

[18] L. Zhou and Z.J. Haas, "Securing Ad Hoc Networks". *IEEE Network*, vol. 13, i. 6, pp. 24-30, November/December 1999.