



HAL
open science

Certification-based trust models in mobile ad hoc networks: A survey and taxonomy

Mawloud Omar, Yacine Challal, Abdelmadjid Bouabdallah

► To cite this version:

Mawloud Omar, Yacine Challal, Abdelmadjid Bouabdallah. Certification-based trust models in mobile ad hoc networks: A survey and taxonomy. *Journal of Network and Computer Applications (JNCA)*, 2012, 35, pp.268-286. 10.1016/j.jnca.2011.08.008 . hal-00644491

HAL Id: hal-00644491

<https://hal.science/hal-00644491v1>

Submitted on 24 Nov 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Certification-based Trust Models in Mobile Ad Hoc Networks: A Survey and Taxonomy

Mawloud Omar^a, Yacine Challal^b, and Abdelmadjid Bouabdallah^b

^a*Université A/Mira, ReSyD, Bejaia, Algérie.*

^b*Université de Technologie de Compiègne, Heudiasyc-UMR CNRS 6599, Compiègne, France.*

Abstract

A mobile ad hoc network is a wireless communication network which does not rely on a pre-existing infrastructure or any centralized management. Securing the exchanges in such network is compulsory to guarantee a widespread development of services for this kind of networks. The deployment of any security policy requires the definition of a trust model that defines who trusts who and how. There is a host of research efforts in trust models framework to securing mobile ad hoc networks. The majority of well-known approaches is based on public-key certificates, and gave birth to miscellaneous trust models ranging from centralized models to web-of-trust and distributed certificate authorities. In this paper, we survey and classify the existing trust models that are based on public-key certificates proposed for mobile ad hoc networks, and then we discuss and compare them with respect to some relevant criteria. Also, we have developed analysis and comparison among trust models using stochastic Petri nets in order to measure the performance of each one with what relates to the certification service availability.

Keywords: Trust, Public-key, Certificate, Security, Mobile ad hoc networks.

1. Introduction

Mobile ad hoc networking [45, 56] is emerging as an important area for new developments in the field of wireless communication. The premise of forming a mobile ad hoc network is to provide wireless communication between heterogeneous devices, anytime and anywhere, with no infrastructure [28, 39, 47]. These devices, such as cell phones, laptops, palmtops, etc. carry out communication with other nodes that come in their radio range of connectivity. Each participating node provides services such as message forwarding, providing routing information, authentication, etc. to form a network with other nodes spread over an area. With the proliferation of mobile computing, mobile ad hoc networking is predicted to be a key technology for the next generation of wireless communications [15]. They are mostly desired in military applications [46] where their mobility is attractive, but have also a high potential for use in civilian applications such as coordinating rescue operations in infrastructure-less areas [10], sharing content and network gaming in intelligent transportation systems, surveillance and control using wireless sensor networks [60], etc.

Inherent vulnerability of mobile ad hoc networks introduce new security problems, which are generally more prone to physical security threats. The possibility of eavesdropping, spoofing, denial-of-service, and impersonation attacks increases [9]. Similar to fixed networks, security of mobile ad hoc networks is considered from different points such as availability, confidentiality, integrity, authentication, non-repudiation, access control and usage control [61, 63]. However, security approaches used to protect the fixed networks are not feasible due to the salient characteristics of mobile ad hoc networks. New threats, such as attacks raised from internal malicious nodes, are hard to defend [11]. The deployment of any security service requires the definition of a trust model that defines who trusts who and how. There are recent research efforts in trust models framework to securing mobile ad hoc networks. There exist two main approaches: (1) Cooperation enforcement trust models [4, 38, 22, 21, 64, 2, 35, 32, 6, 31], and (2) Certification-based trust models [61, 7, 8, 49, 59, 29, 16, 23, 24, 53]. In table 1, we present the major differences between cooperation enforcement trust models and

certification-based trust models.

The first trust models category is based basically on reputation among nodes. The reputation of a node increases when it carries out correctly the tasks of route construction and data forwarding. The models of this category support effective mechanisms to measure the reputation of other nodes of the network. They also incorporate techniques that isolate the misbehaving nodes that are those that show a low reputation value. Trust models based on cooperation enforcement are well surveyed in the literature. Marias et al. provided such a thorough survey of cooperation enforcement trust models in [37]. In this paper, we are interested in the category of certification-based trust models. Indeed, in this category, the trust relationship among users is performed in a transitive manner, such that if \mathcal{A} trusts \mathcal{B} , and \mathcal{B} trusts \mathcal{C} , then \mathcal{A} can trust \mathcal{C} . In this relationship, the principal \mathcal{B} is called Trusted Third Party (TTP). The latter could be a central authority (like CA - Certification Authority) or a simple intermediate user. Both points of view gave birth to two categories of models: (a) Authoritarian models, and (b) Anarchic models. In this paper, we review and classify the existing certification-based trust models belonging to each category. Moreover, to determine the efficiency of a given trust model, it is very important to estimate the certification service availability with respect to mobile ad hoc networks configuration. Therefore, we have modeled the certification process of each surveyed trust model using stochastic Petri nets (SPN) [17, 18]. As you will see in the following sections, this allows a better understanding of the performances of the different models and how to leverage some parameters for higher certification service availability.

While a number of surveys [36, 3, 54] covering the issues of key management in mobile ad hoc networks, have provided some insightful overviews of the different schemes proposed in the literature, none of them focuses on issues related to certificates management thoroughly (the scheme architecture, how the certificates are stored and managed, the complexity evaluation of the certification protocol, etc). To complement those efforts, this work provides detailed taxonomy of certification-based trust models, and illustrates in depth the different schemes by providing the advantages and drawbacks

Table 1: Cooperation enforcement vs. Certification-based trust models

	Cooperation enforcement trust models	Certification-based trust models
Trust degree \mathcal{T}	Variable according to the node's behavior, such $\mathcal{T} \in]0, 1[$.	Decided in a strict manner: trusted or untrusted, such $\mathcal{T} \in \{0, 1\}$.
How to evaluate the trust degree of a new member node?	Supposed as a trusted node, then its trust degree will be updated according to its behavior.	Offline authentication through the policy of certification.
How to evaluate the trust degree of a given node at the first interaction?	Through the recommendation of its neighbor nodes.	Through the certificates chain verification from a trusted party to the node.
Node exclusion	The node will be isolated if the value of its trust degree decreases at a certain threshold.	Through the revocation of its certificate.

of each one with respect to relevant criteria. The careful examination and analysis has allowed us to carry out a comparative study of the proposed schemes based on an analytic evaluation. The ultimate goal of this paper is to identify the strengths and weaknesses of each scheme in order to devise a more effective and practical certificate-based trust models which can achieve a better trade-off between security and performance.

The remaining of this paper is structured as follows. In Section 2, we recall background material relating to basic concepts on cryptography and threshold cryptography. Then, in Section 3, we identify requirements relating to certificates management with respect to mobile ad hoc networks environment and constraints, and in Section 4 we propose a taxonomy of the existing certification-based trust models. Respectively, in Section 5 and 6, we review the authoritarian models, and anarchic models. For each solution, we provide a brief description and discuss its advantages and shortcomings. We model the different solutions using stochastic Petri nets and provide analytical results and conclusions. Then, we make a general analysis and comparison against some important performance criteria. We finally conclude this paper in Section 7.

2. Background

In this section we recall the definition of some security services using cryptographic mechanisms [40, 51].

2.1. *Security Services and Basic Cryptography Mechanisms*

Confidentiality is a service used to keep the content of information from all, but those authorized to have it. Confidentiality is guaranteed using *encryption*. Encryption is a cryptographic transformation of the message into a form that conceals the message original meaning to prevent it from being known or used. If the transformation is reversible, the corresponding reversal process is called *decryption*, which is a transformation that restores the encrypted message to its original state. With most modern cryptography, the ability to keep encrypted information secret is based not on the cryptographic encryption algorithm, which is widely known, but on a piece of information called a *key* that must be used with the algorithm to produce an encrypted result or to decrypt previously encrypted information. Depending on whether the same or different keys are used to encrypt and to decrypt the information. We distinguish between two types of encryption systems used to assure confidentiality:

- *Symmetric-key encryption*: a secret key is shared between the sender and the receiver and it is used to encrypt the message by the sender and to decrypt it by the receiver. The encryption of the message produces a non-intelligible piece of information; the decryption reproduces the original message.
- *Public-key encryption*: also called asymmetric encryption, involves a pair of keys (public and private keys) associated with the sender. Each public-key is published, and the corresponding private-key is kept secret by the sender. Message encrypted with the sender's public-key can be decrypted only with the sender's private-key. In general, to send encrypted message to someone, the sender encrypts the message with that receiver's public-key, and the receiver decrypts it with the corresponding private-key.

Authentication is a service related to identification. This function applies to both entities and information itself. Two parties entering into a communication should identify each other.

The public-key certificate is a digital data structure issued by a trusted third party to certify a public-key's ownership. Among other information a public-key certificate contains: (1) certificate number; (2) issuer's identity; (3) owner's identity; (4) owner's public-key; (5) signature algorithm; (6) period of validity; and (7) the issuer's signature, and eventually other extensions.

CA (Certification authority) is a trusted third party, which is usually a trustworthy entity for issuing certificates. If the same CA certifies two users, then they would have the same CA in common as a third trust party. The two users would then use the CA's public-key to verify their exchanged certificates in order to authenticate the included public-keys and use them for identification and secure communication. Each CA might also certify public-keys of other CAs, and collectively forms a hierarchical structure. If different CAs certify two users, they must resort to higher-level CAs until they reach a common CA (cf. figure 1).

Web-of-trust model [1] doesn't use CAs. Instead, every entity certifies the binding of identities and public-keys for other entities. For example, an entity u might think it has good knowledge of an entity v and is willing to sign v 's public-key certificate. All the certificates issued in the system forms a graph of certificates, named web-of-trust (cf. figure 2).

2.2. *Threshold Cryptography*

A (k, n) threshold cryptography scheme ($k \leq n$) is a cryptographic technique that allows to hide a secret S in n different shares S_i ($1 \leq i \leq n$), so that the knowledge of at least k shares is required and sufficient to recover the initial secret S (cf. figure 3). Let us illustrate this technique with the following famous scheme: Shamir's threshold scheme [50] is based on polynomial interpolation and the fact that a univariate polynomial $y = f(x)$ of degree $k - 1$ is uniquely defined by distinct k points (x_i, y_i) .

Setup. The trusted party T begins with a secret integer $S \geq 0$ it wishes

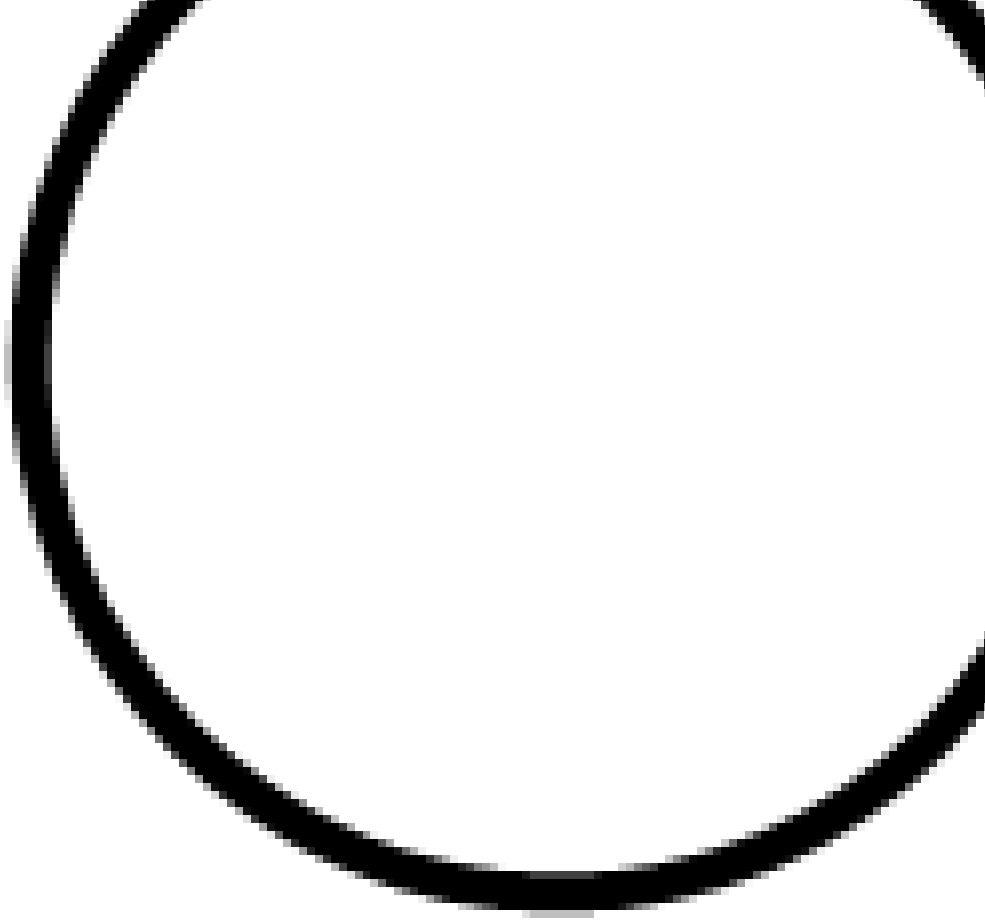


Figure 1: Hierarchical certification authorities

to distribute among n users:

1. T chooses a prime $p > \max(S, n)$, and defines $f(0) = a_0 = S$.
2. T selects $k - 1$ random, independent coefficients a_1, \dots, a_{k-1} , $0 \leq a_j \leq p - 1$, defining the random polynomial over Z_p , $f(x) = \sum_{j=0}^{k-1} a_j x^j$ (where $a_0 = S$).
3. T computes $S_i = f(i) \bmod p$, $1 \leq i \leq n$ (or for any n distinct points i , $1 \leq i \leq p - 1$), and securely transfers the share S_i to user u_i , along with public index i .

Recovering the secret. To recover the initial secret S , a subgroup of at least k users should exchange their shares. After the exchange, each user of the subgroup will get k distinct points (i, S_i) of the polynomial f . These k points allow calculating the coefficients of the polynomial f using the Lagrange interpolation as follows: $f(x) = \sum_{i=1}^k S_i \prod_{1 \leq j \leq k, j \neq i} \frac{x-j}{i-j}$. Since $f(0) = a_0 = S$, the shared secret may be expressed as: $S = \sum_{i=1}^k c_i S_i$, where $c_i = \prod_{1 \leq j \leq k, j \neq i} \frac{j}{j-i}$.

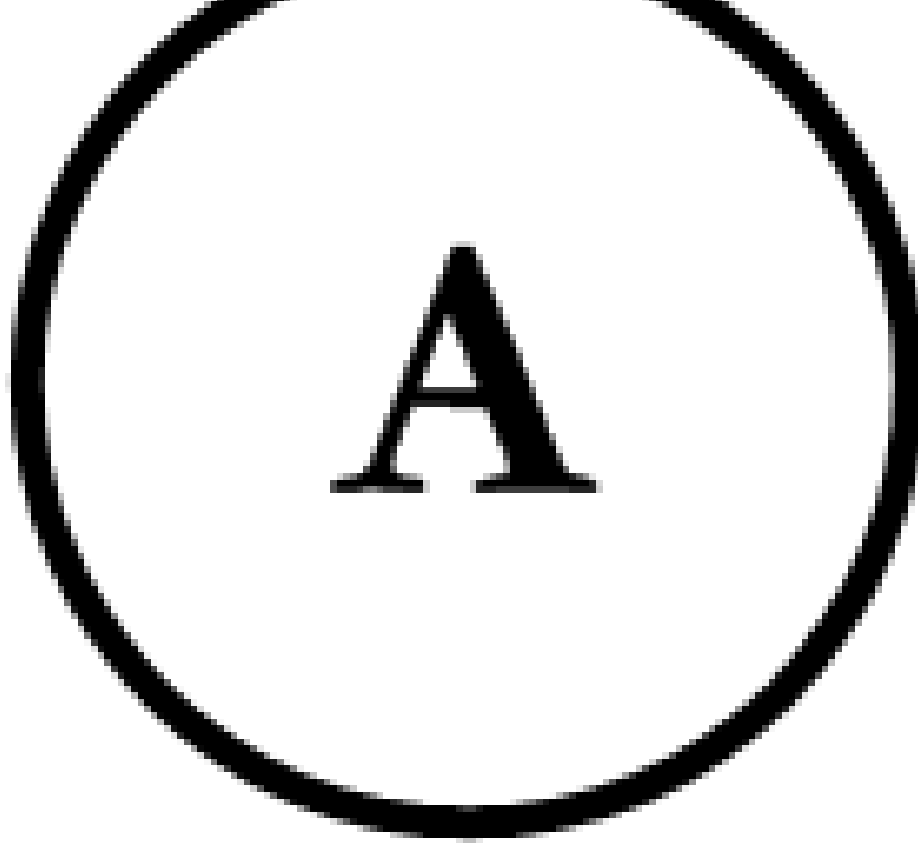


Figure 2: Web-of-trust model

3. Design Issues

The distribution of public-keys and management of certificates have been widely studied in the case of infrastructure-based networks. In the latter, several issues have been well discussed. However, the certificates management in mobile ad hoc networks addresses additional new issues appeared from the constraints imposed, in particular, by the ad hoc network environment. These issues can be resumed in the following points:

Certification service availability issue. In mobile ad hoc networks, due to the frequent link failures, nodes mobility, and limited wireless medium, it is typically not feasible to maintain a fixed centralized authority in the network. Further, in networks requiring high security, such a server could become a single point of failure. One of the primary requirements is to distribute the certification service amongst a set of special nodes (or all nodes) in the network.

Resources consumption issue. Since the nodes in mobile ad hoc network typically run on batteries with high power consumption and low memory capacity, the certification service must be resource-aware. That means the time and space complexity of the underlying protocols



Figure 3: Threshold cryptography

must be acceptably low in terms of computation, communication, and storage overheads.

Scalability issue. Many applications in mobile ad hoc networks involve a large number of nodes. When the certificates management is handled through a centralized authority, the latter may become overloaded due to the number of nodes request. Otherwise, if the certification service is designed in a fully distributed way among several nodes in the network, each participant to the service must maintain a local repository, which contains a maximum number of certificates concerning the other nodes in the network. Hence, the storage overhead will be linear to the network size, which may compromise the system scalability to large ad hoc networks.

Handling heterogeneity issue. As in the case of wired networks, the certifying authorities might be heterogeneous even in mobile ad hoc networks. This means that two or more nodes belonging to different domains (mainly in term of certification policy) may try to authen-

ticate each other. In such a case, there must be some kind of trust relationship between the two domains.

4. Taxonomy

In figure 4, we propose a taxonomy of the existing certification-based trust models for mobile ad hoc networks. We divide existing solutions into two categories depending on the existence or not of central authorities.

4.1. Authoritarian models

In this category, there exist one or more authorities that are trusted by the whole community of ad hoc nodes. Depending on the number of authorities, this category can be further divided into monopolist models and oligopolist models:

1. **Monopolist models.** In this subcategory, the system is ensured by a certification authority. To cope with the spontaneous nature of mobile ad hoc networks, the service is distributed among several servers, which ensure collectively the CA's role using a (k, n) threshold cryptography scheme. The CA's private-key is divided into n private-shares, such that each server holds one private-share. In order to deliver a certificate to a given client node, each server creates a partial certificate (certificate signed using a private-share). The system processes the client request, such that the combination of any k partial certificates gives as a result a valid certificate signed by the CA's private-key. This subcategory is divided into:
 - (a) **Single distributed CA**, where the certification service, in the whole system, is ensured by only one CA, which is distributed among several servers.
 - (b) **Hierarchical CAs**, where the certification service is ensured by several homogeneous CAs organized into a hierarchy. Each or some CAs in the system is distributed among several servers. A trust relationship should be established among the different CAs in this case.

Range

Doit

Figure 4: Taxonomy of certification-based trust models in mobile ad hoc networks

2. **Oligopolist models.** In this subcategory, the system is composed of several heterogeneous CAs. Each CA has its own policy of certification. Each or some CAs in the system are distributed among several servers.

4.2. *Anarchic models*

In this category of models, there is no central authority. Or in other words, each user acts as an authority independently of other users in the network. The propagation of trust in the network forms what is commonly called web-of-trust. As previously outlined, the web-of-trust is managed by users themselves. This model is decentralized in nature, and so very adequate for mobile ad hoc networks. In this category of trust models, two main operations are addressed: (1) the initial web-of-trust construction, and (2) the certificates chains discovery. This subcategory can be further divided into proactive models and reactive models:

1. **Proactive models.** In this subcategory, the protocol of certificates collection is executed systematically among neighboring nodes. Thus, when the node needs to verify a certificate, it is done instantly since the required chain of certificates would have been already retrieved from the network.
2. **Reactive models.** In this subcategory, the certificates collection protocol is executed on-demand. When the node needs to verify a certificate, it collects in a distributed manner the appropriate chain of certificates from the network. This prolongs the delays of certificates verification.

In the following sections, we give detailed descriptions of certification-based trust models belonging to each category. We give for each trust model an overview, advantages, drawbacks, and eventually the proposed extensions. Then, for each category, we give an analytical modeling and an overall comparison with respect to the criteria presented in Section 3.

5. **Authoritarian Models**

In this section we present and discuss certification-based trust models belonging to the authoritarian models category.

5.1. *Monopolist Models*

In this class of trust models, the certification service is ensured by a single or several homogeneous CA.

5.1.1. *Single Distributed CA*

In this subclass of trust models, the certification service is ensured by a single CA, which is distributed among several servers.

Zhou and Haas

Overview. Zhou and Haas [61] explored the issue of distributed CA in mobile ad hoc network, with the assumption of a single authority domain across the network. They proposed a partially distributed CA relying on a (k, n) threshold cryptography scheme. This work is the first to introduce the threshold cryptography into the security protocols in mobile ad hoc networks. This work provides an excellent guide to the following works. Their objective was to distribute the trust among nodes of the network such that no less than a certain threshold of nodes are trusted. The CA is distributed among particular nodes, called *servers*. For the service to sign a certificate, each server generates a partial signature for the certificate using its private-share and submits the partial signature to a combiner. With k correct partial signatures, the combiner is able to compute the signature for the certificate. We describe in figure 5 the global functioning of their scheme. First, a requester node contacts a combiner server. The combiner replicates the request to the other servers, and each one generates a partial certificate and returns it to the combiner in order to construct the complete certificate, and finally, forward it to the node. To improve security, they also proposed to use proactive share update to compute a new set of private-shares after a certain time interval.

Advantages and Drawbacks. With this scheme, even if an adversary discovers the private-shares of some, but less than k servers, it still cannot recover the CA's private-key. Moreover, this scheme allows only some selected nodes the ability to serve as part of the CA, and thus take part in admission decisions. Furthermore, this scheme does not describe how a node can contact k servers securely and efficiently when the servers are scattered in a large area, how to keep the n special

Compl

Figure 5: Zhou and Haas ($k = 3, n = 6$)

nodes available when needed, and how the nodes know how to locate the servers. All of this makes the maintenance of the system complex. Also, it is not explained how to ensure the secure distribution of private-shares at the share refreshing phase. Moreover, the problem of availability was not addressed. Indeed, the right choice of the threshold value k , which is a trade-off between availability and resilience is not discussed. If the threshold value is large, the availability will be decreased but this increases the robustness and vice versa.

MOCA - Yi and Kravets

Overview. Yi and Kravets [59] proposed a distributed CA, ensured by selected nodes as MOCA servers (MOBILE Certification Authority). In this scheme, two main criteria have been involved: the selection and the maintenance of MOCA servers. In this scheme, MOCA servers are selected based on the heterogeneity aspect among existing nodes

in the network. ¹ Physically more secure and computationally more powerful nodes are the typical choices for MOCA servers. These selected servers share the private-key and collectively provide the CA functionality. All nodes are equipped with MP (MOCA certification Protocol), which enables communication with MOCA servers. In the initial version of MP, nodes flood the network with certification requests and MOCA servers that receive the request respond with a certification reply. The flooding technique has a high overhead, and to alleviate it, they investigated the cache tables of client nodes and discovered that with a certain amount of certification traffic in the network, a mobile node tends to have many cached routes entries to enough MOCA servers. Then, they exploit these caches to reduce to broadcasting overhead.

Advantages and Drawbacks. Compared to the scheme of Zhou and Haas, this one limits the candidates who share the CA's private-key, to whom the CA service is assigned, to secure and powerful nodes in the network. Therefore, the scheme became efficient and robust. However, they left open an important question: how and who judge the level of security in choosing MOCA servers? They also proposed a new pattern of communication, termed as "manycast", between a client node and MOCA servers. The pattern is based on a strong assumption that each node knows which nodes in the network are MOCA servers and their positions.

Dong et al.

Overview. Dong et al. [13] investigated also the problem of CA servers localization, and proposed a CA cluster-based architecture. The system organizes mobile ad hoc network into clusters². Each cluster head

¹For example, in a military battle field scenario (one of the most mobile ad hoc network's application domain), there can be many different types of mobile nodes in the field (e.g. infantry soldiers, tanks, platoon leader's jeeps, command and control vehicles, etc).

²For more information about clustering architectures in mobile ad hoc networks,

(CH) maintains a CA servers information table, which contains a list of CA nodes in its local cluster and eventually in the other clusters. The procedure of certification is performed as follows. When a client node u_i requires a certificate, it sends a request to its cluster head CH_i to get information about local CA servers in its cluster. The CH_i collects distributed CAs information, and forwards them to u_i . Then, u_i selects k CA servers according to information provided by CH_i , and sends the certification request to them. The collaborative certification is handled by each CH_i , which combines the received k partial certificates and generates the complete one. If the number of CA servers is less than k (the threshold value), the CH_i solicits the CA servers in the other clusters. Thus, CH_i sends a request message to all other CHs, and each CH receiving the request message responds with a message indicating the number of CA servers in its cluster. Then, u_i selects what it lacks and submits to them its certification request.

Advantages and Drawbacks. This scheme achieves a well service flexibility, which CA information are managed among CHs, which reduces service response delay and system overhead. However, authors have given more interest to CA servers availability criterion. An important criterion which was neglected in particular is the CA's correctness degree which be available within clusters. If there are enough CA servers within a given cluster, among which there are some ones compromised, how avoid them? Is there a mechanism to request other CA servers from the other clusters?

Kong et al.

Overview. Kong et al. [27] proposed another style of threshold cryptography based trust model by distributing the CA's private-key to all the nodes in the network, contrary to the previously presented schemes. In other words, each normal node holds a private-share, and multiple

see [58].

Reque

nodes
servic
to be
centra
least j
to get
partia
is a tr
bined
Anoth
a priv
alread
functi

Partial

Figure 6: Kong et al. ($k = 3, n = 16$)

Advantages and Drawbacks. Compared to the previous schemes, this one is fully distributed, where the CA's role is distributed among all

nodes in the network. Therefore, this solution is better since it is easier for a given node to locate k neighbor nodes and request them since they are part of the CA service. On the other hand, it minimizes the effort and complexity for mobile nodes to locate CA service providers. One of the two major weaknesses of this scheme is that it is difficult to set an appropriate threshold k , which is a globally fixed parameter that is honored by each entity in the system. Another problem is that the scheme permits the new nodes to obtain their private-shares from a threshold number of member nodes. However, this property has a drawback: an adversary could take as many identities as necessary (*Sybil attack* [12]) to collect enough private-shares, and thereby construct the CA's private-key.

Extensions and Improvements. Luo et al. [30, 33] proposed extensions to the scheme of Kong et al. In particular, the proposal involves a framework for parallel private-share updates, and improves the CA service by a certificate revocation mechanism. The parallel private-share updates builds on the scheme of Herzberg et al. [19]. However, unlike the latter, which requires each node to collect inputs from all the other nodes before its new private-share can be computed, this scheme stipulated that firstly a coalition of k nodes update their private-shares using the scheme of Herzberg et al.; then the coalition of k nodes can update the private-shares of the remaining nodes utilizing the self-initialization scheme employed in the scheme of Kong et al. This therefore allows parallelization, and consequently a more efficient private-share update process. The certificate revocation mechanism can be briefly described as follows. Each node u maintains a certificate revocation list (CRL). An entry in the CRL consists of an accused node's identity and a list of the node accusers. If a node accuser list contains less than k legitimate accusers, the node is marked as *suspect*. Otherwise, the node in question is considered by node u to be misbehaving or compromised, and is marked as *convicted*. A node can also designate a neighboring node v as been convicted if by its observation u deems v to be misbehaving or compromised. In such case, u broad-

casts an accusation against v . When a node u receives an accusation against any given node, u first checks if the accuser is a convicted node in its CRL; if it is, the accusation is discarded; otherwise, it updates its CRL with the relevant information. When a node is delineated as being convicted, it is removed from all accuser list. A convicted node is re-classified as being suspected if its number of accusers falls below k .

Raghani et al.

Overview. Raghani et al. [49] proposed a fully distributed CA, where the CA's role is performed by the neighboring nodes as the scheme of Kong et al. In order to maintain the highest availability of the CA's services, they suggested that the threshold value must be equal to the average node degree (number of one hop neighbors). Based on this consideration, they discussed the importance of the threshold parameter value and their impact on the service availability, and then, proposed a certification-based trust model using a dynamic threshold cryptosystem. The latter allows to dynamically adjusting the value of the threshold when required by monitoring the average node degree of the network. Periodically, each node executes the neighbor discovery protocol in order to calculate the degree of this node. On obtaining the response from neighbors, each node calculates the degree value and communicates it to a special node, called *leader node*. The latter uses the degree values to calculate the average node degree of the network. When the average node degree is less than the threshold value, certification requests would fail. In order to prevent this, the leader node initiates a change in the threshold value when the average node degree of the network falls below the current threshold value. The new value of the threshold is calculated as follows: $k_{new} = \max(k_{min}, \frac{9d}{10})$, where $\frac{9}{10}$ is a multiplicative factor, k_{min} is the minimum value of the threshold (authors suggested that $k_{min} = 2$), k_{new} is the new threshold value, and d is the average node degree of the network. The multiplicative factor must be less than or equal to one and depends on the security requirements of the application. The use of a multiplicative factor less

than one prevents the frequent change in the threshold value when the number of nodes continuously goes down in a network. When the leader node decides to change the threshold value, it sends the new threshold value to all the nodes. Then, nodes begin to update their private-shares corresponding to the new threshold value.

Advantages and Drawbacks. Compared to the scheme of Kong et al., this one achieves efficient certification service availability, where the threshold k is adapted according to the available one-hop neighbor nodes. In the practical situations, a change in the threshold value is required when the number of nodes in the network changes. Due to the change in the number of nodes, performance of the service may get affected and thereby requires a change in the threshold value. With large networks, nodes have more chances to locate a sufficient number of neighbors node to satisfy their requests. However, these chances decrease in the small networks. This requirement is well discussed in this scheme. However, there exist other important cases of necessity to change the threshold value, such as when the security requirements of the network change, or when the malicious nodes number evolves in the network. Unfortunately, this direction is neglected in this scheme.

5.1.2. Hierarchical CAs

In this subclass of trust models, the certification service is ensured by a hierarchy of several homogeneous CAs.

DICTIONARY - Luo et al.

Overview. Luo et al. [29] considered a mixed network³ and proposed DICTATE (Distributed CerTification Authority with probabilisTic frEshness for ad hoc networks). DICTATE is a hierarchical trust model between a central CA (mCA - mother CA) in the infrastructure-based portion of the network, and a group of distributed CAs (dCA - distributed CA) in the mobile ad hoc portion of the network. The CA's private-key in the mobile ad hoc network party is shared among dCA servers using threshold cryptography. Indeed, nodes in mobile ad hoc network can collectively be isolated from the mCA server, but always have the need to the certification service. The mCA delegates the dCA servers during the isolation period in order to ensure the availability of security services. Also, the mCA server controls the admission of nodes (either dCA servers or normal nodes) to the system through the issuance of public-key certificates. When the mobile ad hoc portion of the network is disconnected from the mCA server, client nodes submit their requests to the dCA servers. Moreover, this scheme applies an identity-based scheme [44, 5] and a corresponding signature scheme for this public-key pair and make the identities of all dCA servers publicly known, such that each node could calculate the public-key of each dCA server from only its identity. We describe in figure 7 the global functioning of DICTATE.

Advantages and Drawbacks. Compared to the previous works, Luo et al. proposed in this scheme a practical solution, where the problem of distribution of private-shares to the distributed CAs and issuance of the public/private-key pairs to nodes are ensured by the mCA server. However, the scheme is not adapted for pure mobile ad hoc networks. The scheme still necessitates a central administration infrastructure and the access toward the infrastructure-based network. In other hand, the sensitive point in this scheme is the mCA server, which ensures

³A mobile ad hoc network connected to an infrastructure-based network access points.

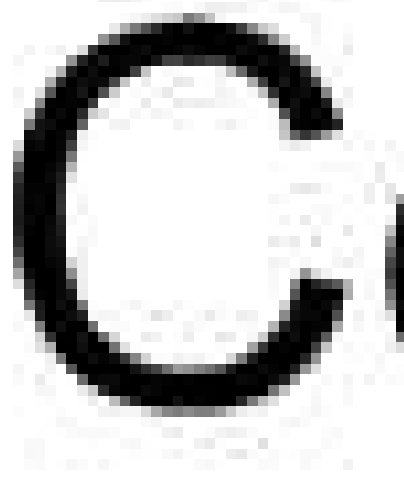


Figure 7: DICTATE, Luo et al.

the whole security services. They discussed well the robustness of the scheme against compromised dCA servers and agents which handle the nodes requests, but the case of a compromised mCA has not been studied.

Extensions and Improvements. Omar et al. [42] proposed NetTRUST (mixed NETworks Trust infrastRUcture baSed on Threshold cryptography); an ameliorated version of DICTATE for mixed networks. The scheme uses two particular CAs, which ensure the certification service: the Central CAs (CCAs) are tied to the portion of infrastructure-base network, and the Mobile CAs (MCAs), which are in mobile ad hoc network party. The MCA servers emulate the CA's role using a (k, n) threshold cryptography scheme, and the CCA servers delegate the role

of certification to the MCA servers using a (t, m) threshold cryptography scheme (cf. figure 8). This scheme includes two threshold cryptosystems: CTCS (Certification Threshold Cryptography System) and DTCS (Delegation Threshold Cryptography System). The CTCS is executed among MCA servers in order to ensure the certification service in the mobile ad hoc network party. The DTCS is executed among CCA servers to ensure the delegation service (using attributes certificates) of novel MCA servers if necessary. Compared to DICTATE, instead to employ a central delegation service in the infrastructure-based portion of the network, NetTRUST ensures the delegation service in a distributed way via another threshold cryptosystem, which increase more the robustness of the system. Another efficient style of delegation is proposed by Ge et al. [16] without being based on the infrastructure-based network. The certification service includes a group of distributed servers that have the role to delegate other servers in the mobile ad hoc network itself. If necessary, some normal nodes are selected and converted to auxiliary servers by the original servers, and they are activated later. When a specific certificate needs to be issued in an isolated segment of the network, where there are not enough servers, some auxiliary servers will be activated. Once activated, the auxiliary server behaves like a normal server in terms of capacity of certification. When the service of an auxiliary server is no longer needed, it will be deactivated by the original servers.

Seys and Preneel

Overview. Seys and Preneel [52] proposed a hierarchical trust model including several levels of CAs instead of two levels, compared with the schemes of DICTATE, NetTRUST, and Ge et al. The certification service is achieved using a (k, n) threshold cryptography scheme at each level of the hierarchy. In each level i , a certification private-key SK_i is shared among n nodes. On the top layer of the hierarchy a master private-key SK_0 is used to issue public-key certificates to the nodes in the level 1. Next to this, all nodes on level 1 share the layer 1's

Figure 8: NetTRUST, Omar et al.

private-key SK_1 and issue public-key certificates to the nodes in the level 2. Similarly, nodes in level 2 receive certificates signed by SK_1 and share the layer 2's private-key SK_2 to issue public-key certificates to the next level. This process is continued until the root levels in the hierarchy (cf. table 2). If a node at some level requires a certificate, it will contact k nodes of the previous level to gather partial certificates and combine them to compute the complete one.

Advantages and Drawbacks. Compared to the previous schemes, this one achieves a better robustness, where each level in the hierarchy is configured by a threshold cryptosystem. However, the drawback of

Table 2: Seys and Preneel

Layer	CA's private-shares	Certificates
level 0	$SK_0 = (s_1^0, \dots, s_n^0)$	–
level 1	$SK_1 = (s_1^1, \dots, s_n^1)$	$C_0(PK_{1,1}) \dots C_0(PK_{1,n})$
level 2	$SK_2 = (s_1^2, \dots, s_n^2)$	$C_1(PK_{2,1}) \dots C_1(PK_{2,n})$
\vdots	\vdots	\vdots
level h	–	$C_{h-1}(PK_{h,1}) \dots C_{h-1}(PK_{h,n})$

this scheme is the computational overhead⁴. In order to verify one chain of certificates according to h levels, the node must calculate h complete certificates, at the cost of a higher computational overhead.

5.2. Oligopolist Models

In this class of trust models, the certification service is composed of several heterogeneous CAs, which each one has its own policy of certification.

Wang et al.

Overview. Wang et al. [55] proposed a distributed certification-based trust model of heterogeneous CAs. It means that the network contains several distributed CAs; each one of them share the private-key of a one CA. In order to handle heterogeneous CAs, each client node maintains a list of CAs that it trusts. When a client node requires to authenticate another node, they start by exchanging CAs lists. Then, they compare both lists to check if there are some common ones, and if so, they exchange their certificates signed by this common CA. Otherwise, they try to search in their one-hop and two-hop neighbors. The mutual authentication protocol of two nodes u and v is performed as follows. The client node u sends a certificate request to v , which contains currently trusted CAs list. In return, v sends back also its trusted CAs

⁴The complexity of computational overhead of partial certificates combination according to a (k, n) threshold cryptography scheme is: $5n + k^2 + 4k + 5$ [20].

list. Then, v compares the two lists. If there exists a common CA that both nodes trust, v then sends to u its certificate signed by this CA, and in return, receives the u 's corresponding certificate. Otherwise, v would attempt to find a CA that it may deem trustworthy and is trusted by u , in order to authenticate it. The search is done by finding a set of v 's one-hop and two-hop neighbor nodes. If this set size reaches k (the threshold value), the corresponding CA is deemed to be trustworthy to v . Then, v selects k shareholders from this set and request them for the u 's certificate. Otherwise, if v cannot find enough partial certificates, the authentication procedure performed by v fails.

Advantages and Drawbacks. Compared to the other schemes, this scheme has the advantage to allow multiple heterogeneous distributed CAs systems to coexist in the network; whereas, all the trust models reviewed above facilitate the system architecture by a single distributed CA in the whole network. However, authors did not explain why the authentication protocol uses up one-hop and two-hop neighbors nodes. Increasing the number of helper nodes will surely increase more chances in finding enough common CAs, instead of confirming that the authentication will fail.

Xu and Iftode

Overview. Xu and Iftode [57] proposed a locality driven certification-based trust model. They envision a mobile ad hoc network as a group of interacting subnetworks. Each subnetwork establishes a distributed CA using threshold cryptography. A distributed CA issues certificates to nodes in its subnetwork and provides public-key authentication services for its community. Each subnetworks's CA includes a special node (named *dealer*), which is trusted by all nodes. It is initialized by this dealer in order to generating the CA's public and private-keys, and to distribute private-shares. In addition, the proposed scheme assumes that there exist trust relationships among CAs of each subnetwork. These trust relations are utilized to authenticate certificates

issued
foreign
maintain
tory.

Figure 9: Xu and Iftode

Advantages and Drawbacks. As the scheme of Wang et al., this one has the advantage of handling heterogeneous CAs by establishing trust relationships among them. The main weakness of this scheme is the commonly trusted dealer in each community.

5.3. Modeling and Discussion

In order to measure the degree of the possibility to get a successful certification process, we have opted to model trust models using SPN (Stochastic Petri Network) [17]. This model is adequate in the sense that the availability of servers at a given moment for a given node requester is probabilistic

Figure 10: SPNs of authoritarian models

and depends on many parameters such as mobility, nodes availability, radio links failure, etc. Then, the servers must collaborate collectively to generate a public-key certificate which requires the synchronization of at least k servers. Indeed, SPNs consist of places and transitions as well as a number of functions. Enabled transitions fire according to exponential distributions; characteristic of Markov Processes. It allows the quick construction of a simplified abstract model that is numerically solved for different model parameters. In figure 10, we present SPNs corresponding to each trust model

belonging to this category, and we note in table 3 the most used terminology in this subsection.

Table 3: Notations

Term	Description
λ_{AReq}	Interarrival duration rate of authentication requests.
λ_{AS}	Interarrival duration rate of servers.
λ_{AN}	Interarrival duration rate of neighboring nodes.
T_S	The transition representing the certification service execution.
$P\{T_S\}$	The probability of crossing the transition T_s in the SPN.
μ	The certification service duration.

5.3.1. Successful Certification Probability Calculation (Zhou and Haas Scheme)

For instance, in what relates to the scheme of Zhou and Haas, for a given certification request, the operation process needs the availability of k servers. Therefore, in order to get a successful certification process, there are two conditions: (1) a certification request arrives (with rate λ_{AReq}), and (2) k servers should be accessible (with rate λ_{AS}). The corresponding SPN is illustrated in figure 10 (a). Each transition T_i is enabled through a stochastic process with an average rate λ_i . At the arrival of a certification request, the transition T_{AReq} will be reached, which means that the certification request is ready to be executed, and thereby the next place will be configured by a token. The certification process requires the availability of at least k servers. If one of the n servers is available, the transition T_{AS} will be reached, and a token will be added in the next place. The availability of k servers allows to cross the transition T_{AS} k times and hence the next place will be set by k tokens, which allows executing the certification process, which is represented by the transition T_S (accorded to the rate $\lambda_S = 1/\mu$). The probability of crossing the transition T_S means the probability of successful certification. This probability can be calculated as follows [34]: $P\{T_S\} = \frac{\lambda_S}{\lambda_{AReq} + k\lambda_{AS} + \lambda_S} = \frac{\frac{1}{\mu}}{\lambda_{AReq} + k\lambda_{AS} + \frac{1}{\mu}} = \frac{1}{\mu\lambda_{AReq} + k\mu\lambda_{AS} + \frac{\mu}{\mu}} = \frac{1}{\mu\lambda_{AReq} + k\mu\lambda_{AS} + 1}$. Therefore, we conclude that there are mainly two param-

ters that influence the probability $P\{T_S\}$: k and λ_{AS} . The threshold value k is a cornerstone parameter influencing the overall performance of the certification service. Indeed, the greater is the value of k , the smaller is the probability of successful certification $P\{T_S\}$. This can be interpreted by the fact that if k increases, the client node must solicit a large number of servers, which decreases the certification service availability. On the other hand, if λ_{AS} increases, the probability of successful certification $P\{T_S\}$ decreases. Indeed, λ_{AS} represents the degree of availability of the servers in the system. For example, if the system includes a large number of servers, the interarrival duration rate of servers decreases, and then the probability $P\{T_S\}$ increases. Hence, the value of λ_{AS} is tightly related to the number of servers (n) and the servers selection policy adopted in the trust model.

5.3.2. Overall Analysis

In table 4, we provide a comparison of the different authoritarian trust models with respect to the rate of successful certification $P\{T_S\}$. For each model, we provide the calculus of $P\{T_S\}$ according to their respective SPNs depicted in figure 10, and present how are selected the servers and the threshold value k ; the main influencing parameters on the rate of successful certification $P\{T_S\}$. In this category, we remark that the probability of successful certification is of the form: $P\{T_S\} = (\mu\lambda_{AReq} + \alpha k\mu\lambda_{AS} + 1)^{-1}$, where α is a constant whose value depends on the trust model. The constant α equals to 1 for the most trust models. This does not mean that the different trust models have all the same successful certification rate. Indeed, the latter depends on the values of λ_{AS} (the way servers are selected), and the threshold value k . Depending on the considered trust model, the values of these two parameters belong to some different intervals dictated by the trust model nature and design. This allows further classifying the trust models of this category into four classes of performance (with respect of successful certification rate) depending on the intervals in which evolve λ_{AS} and k . The plot of $P\{T_S\}$ in figure 11 shows these classes.

Class A: To this class belongs the scheme of Raghani et al. [49], which ensures a high level of certification service availability. In this trust model, the

Table 4: Comparison with respect to the successful certification rate $P\{T_S\}$, where k is the threshold value, n is the number of distributed servers, h is the number of levels in the CA hierarchy, and d is the average of neighboring nodes in the network.

Scheme	Servers	Threshold	$P\{T_S\}$	Quality of $P\{T_S\}$
Zhou and Haas	n is static.	Static, $k < n/3$.	$(\mu\lambda_{AReq} + k\mu\lambda_{AS} + 1)^{-1}$	Medium
MOCA	n is static. Moreover, servers to select must be better physical security or processing capability.	Static.	$(\mu\lambda_{AReq} + k\mu\lambda_{AS} + 1)^{-1}$	Medium
Dong et al.	n is static.	Static.	$(\mu\lambda_{AReq} + k\mu\lambda_{AS} + 1)^{-1}$	Medium
Kong et al.	n is dynamic, such each normal node is a server.	Static.	$(\mu\lambda_{AReq} + k\mu\lambda_{AN} + 1)^{-1}$	High
Raghani et al.	n is dynamic, such each normal node is a server.	Dynamic, $k = \max(2, \frac{9d}{10})$.	$(\mu\lambda_{AReq} + \frac{9d}{10}\mu\lambda_{AN} + 1)^{-1}$	High
DICTATE	n is dynamic, such auxiliary dCA servers may be added dynamically by the mother CA.	Static, $k < n/3$.	$(\mu\lambda_{AReq} + k\mu\lambda_{AS} + 1)^{-1}$	High
NetTRUST	n is dynamic, such auxiliary MCA servers may be added dynamically by CCA servers.	Static.	$(\mu\lambda_{AReq} + k\mu\lambda_{AS} + 1)^{-1}$	High
Ge et al.	n is dynamic, such auxiliary servers may be added dynamically by original servers.	Static.	$(\mu\lambda_{AReq} + k\mu\lambda_{AS} + 1)^{-1}$	High
Seys and Preneel	n is static at each level of the hierarchy. However h is dynamic.	Static at each level.	$(\mu\lambda_{AReq} + hk\mu\lambda_{AS} + 1)^{-1}$	Low
Wang et al.	n is static.	Static, such that each pair of nodes must have at least one common CA, i.e. at least k common servers.	$(\mu\lambda_{AReq} + k\mu\lambda_{AS} + 1)^{-1}$	Low
Xu and Iftode	n is static. Moreover, servers to select must be most computing resources or better network connectivity.	Static, $k > n/2$.	$(\mu\lambda_{AReq} + k\mu\lambda_{AS} + 1)^{-1}$	Medium

CA's role is ensured by the neighboring nodes. This decreases the value of λ_{AS} (λ_{AN} in the case of this trust model). Moreover, the second advantage of this trust model is that the value of k is dynamic, which is recalculated periodically and adjusted to the network degree d , so that the certification service availability remains high. This property means that the Raghani et al. scheme is not sensitive to the variation of the value of k , in contrast to the other schemes whose certification service availability depends on the value of k .

Class B: In this class, we find, in the top, the scheme of Kong et al. [27], which ensures a high level of certification service availability. This is due to the fact that all the nodes of the network are involved as servers providing the distributed certification service. This property decreases the value of λ_{AS} . In the second level, we find the scheme of Ge et al. [16], which ensures, also, a high level of certification service availability. In this scheme, the CA's role is ensured by n servers, where auxiliary servers could be dynamically delegated (by original servers in mobile ad hoc network) in the isolated portions of the network. This reduces the value of λ_{AS} . We can consider that, in this scheme, the value of n is dynamic. We find also in this class, the schemes of DICTATE [29] and NetTRUST [42]. However, in these trust models, the new servers, to be added, must be accessible to the delegating servers that are located in the infrastructure-based portion of the network.

Class C: In this class, we find the trust models that ensure a medium level of certification service availability. In this class, the number of servers n is static, which increases the value of λ_{AS} . The trust models of Zhou and Haas [61], MOCA [59], Xu and Iftode [57], and Dong et al. [13] belong to this class. With respect to λ_{AS} , the trust models of MOCA, and Xu and Iftode are less efficient because of the constraint of servers selection, which increases λ_{AS} .

Class D: In this class, we find the trust models that ensure a low level of certification service availability due to the fixed number n of servers. Moreover, the complexity of the certification protocol reduces the cer-

tification service availability. Indeed, Seys and Preneel’s trust model [52] includes a hierarchical CAs of h levels, where each one uses a (k, n) threshold cryptography scheme. Therefore, to perform the certification process, k partial certificates must be combined at each level, so in totally $h \times k$ partial certificates should be combined. Hence, to collect them, the client node must solicit a large number of servers, which increases greatly the value of λ_{AS} and hence reduces the value of the success certification rate $P\{T_S\}$ (cf. table 4). The scheme of Wang et al. [55] belongs also to this class. In this trust model, the system is distributed among N heterogeneous CAs, where each node trusts a subset ζ_i of n_i servers ($n_i \leq N$). To ensure that each client node c_i can be authenticated by another client node c_j , it is necessary that $|\zeta_i \cap \zeta_j| \leq k$. Therefore, another constraint appears for this trust model: in order to ensure a significant successful certification process, the requester node must collect the partial certificates from k servers that it trusts and all the other nodes trust. This constraint increases the value of λ_{AS} , and consequently reduces the probability $P\{T_S\}$.

5.4. Overall Comparison

In table 5, we give an overall comparison of trust models belonging to this category. We summarize the main lessons learnt from this comparison in the following points:

- The availability of the certification service depends on the nodes ability to reach the servers. This property is strongly related to: (1) how to choose servers? (2) how many servers? and (3) how to choose the threshold value of the used threshold cryptography scheme? If the choice of servers is strict (for example, in terms of storage or/and computational capacities), the number of servers will be limited, and so the certification service will be less available. Moreover, the choice of the threshold value influences deeply the certification service availability. If the threshold value is small or dynamically adjustable according to the access capacity of nodes, it can achieve a high level of availability.

- In this category, the most required resource is computation. The latter would be used in the complex calculus induced by the public-key and threshold cryptography algorithms. Moreover, there are also the storage overhead in order to maintain the stock of certificates at each server (or at each node, according to the protocol). The communication overhead is caused by the important number of certificates transmitted in the network. Indeed, in order to transmit only one certificate, k partial certificates must be transmitted.
- The scalability of the certification service depends strongly on the number of distributed CAs. This criterion is important in the sense that the service will be shared among an important number of CAs, instead of surcharging a single one on a large scale. Moreover, the certification service must keep open the possibility of adding auxiliary CAs if needed (like hierarchical models). Another acceptable approach consists of distributing the certification service among all the nodes in order to eliminate the shortcomings of centralized dependency. Therefore, it is possible to achieve a good level of scalability if nodes do not store all the generated certificates in the network.
- Handling the heterogeneity of certification is ensured only in some trust models of this category. The principal objective is to put a bridge among the different CAs (using, for example, a web-of-trust among CAs).

6. Anarchic Models

In this section we present the certification-based trust models belonging to the anarchic models category.

6.1. *Proactive Models*

In this subcategory, the protocol of certificates collecting is executed periodically among nodes. When nodes need to verify a chain of certificates, it collects them in their its locally.

Table 5: Global comparison

Scheme	Availability	Resources Consumption	Scalability	Heterogeneity Support
Zhou and Haas	Medium	Servers store all issued certificates.	No	No
MOCA	Medium	MOCA servers are computationally powerful. The rest of nodes maintains only a table of cached routes to servers.	No	No
Dong et al.	Medium	Each cluster-head maintains information about all CA servers.	No	No
Kong et al.	High	High storage overhead in order to maintain certificates at each node.	Yes	No
Raghani et al.	High	High communication overhead due to the broadcasting of threshold values, which is followed by updating the nodes private-shares.	Yes	No
DICTATE	High	Each node maintains identifiers of the dCA servers (public-keys are computed from the identifiers).	Yes	No
NetTRUST	High	Each node maintains the delegation certificates of MCA servers.	Yes	No
Ge et al.	High	The computational overhead is expensive in creating auxiliary servers private-shares.	Yes	No
Seys and Preneel	Low	The computational overhead is expensive. Certificates verification involves the combination of a large number of partial certificates.	Yes	No
Wang et al.	Low	Each node maintains a list of its trusted CAs.	Yes	Yes
Xu and Iftode	Medium	Each CA maintains a table of trust relations with the other CAs without stores them on local repository.	Yes	Yes

Capkun et al.

Overview. Capkun et al. [7, 8] assumed that "small world" phenomenon found in social relationships applies also to mobile ad hoc network's users relationships, and proposed a fully self-organized trust model that requires no central authority. They used a PGP-like [62] mechanism to initialize the system. This scheme assumes that trust establishment is coming from offline trust relationships, which are generated

from general social relationships. Certificates of all nodes are created by the nodes themselves; precisely by the corresponding user⁵. Each node keeps a certificate repository. If a user u believes that a given public-key belongs to another user, then u can issue a certificate for the given public-key of the user v by signing it. This scheme is represented as a directed certificate graph, where vertices denote users public-keys and edges denote certificates (cf. figure 12). If user u issues a certificate for the user v , then there is a directed edge from K_u to K_v (K_u and K_v are the u 's public-key and v 's public-key, respectively). If there is a directed path from K_u to K_v , the path is the certificate chain and it represents that u believes that K_v belongs to v through some other users. When user u wants to verify user v 's public-key, they merge their local certificate repositories and find an appropriate certificates chain from K_u to K_v in the merged repository. If such chain is not found, u can solicit neighboring nodes in one or two hops (named, helper nodes).

⁵They assumed that each user owns a single mobile node. Hence, the same identifier for the user and her node is used.

Advantages and Drawbacks. The scheme has the advantage of the autonomy, where no central authority is required to assure the security services. Moreover, due to the small world phenomenon shown in social relationships, nodes can thus authenticate each other with acceptable length of trust relationship chains. However, the probability of finding such a certificate chain in this scheme is high, but is not guaranteed. Also, the verification of such chain may be computationally expensive depending on the chain length. The certificate graph, which emerges from this web-of-trust relationship, may not be strongly connected, especially in mobile ad hoc network, and in an extreme case may be partitioned into disconnected components. In this case, nodes within one component may not be able to communicate with ones in other components. Moreover, each end-user is required to build its local certificate repository before it can use the system; this leads to some overhead, both in terms of time and bandwidth.

Ren et al.

Overview. Ren et al. [48] proposed a modified version of the scheme of Capkun et al. by introducing a boot server to initialize the system. The boot server computes and distributes to each node a short list of l bindings (nodes identifiers and public-keys). Then, each of them stores it locally and generates the corresponding certificates.⁶ Thus, a web-of-trust relationships is formed, and the system becomes fully distributed, where nodes authenticate themselves through certificates chains. If a node u gets the binding corresponding to node v and its public-key in the short list, then the binding of node u to its own public-key is also included in the short list of node v . When user u wants to verify user v 's public-key, they exchange their short lists and try to find a certificate chain from u to v .

Advantages and Drawbacks. Compared to the scheme of Capkun et al., this one establishes sufficient trust relationships with minimum local

⁶Lists are not assumed to include the same list of bindings.

storage overhead at mobile nodes. However, this density of relationships depends on the short lists length l . When l equals to 1, then the possibility for any two member nodes to establish a mutual authentication process is extremely low. When l equals to $m - 1$ (m is the network size), we can see that all the nodes are fully connected. However, every node is required to store the bindings of all the m current member nodes (including its own), and therefore, the storage overhead is linear to the network size. Thus, the value of l must be carefully chosen. Also, even if the system is fully distributed, it remains dependent on the boot server. Therefore, this scheme suffers from a strong assumption relating to the availability of such a boot server.

Omar et al.

Overview. Omar et al. [43] proposed a new style of web-of-trust in order to produce a fully distributed trust model for mobile ad hoc networks. The scheme allows nodes to generate, store, and distribute their public-key certificates without any central server or trusted party. Like the previous schemes, in this one, users public/private-keys are created by the users themselves, and key authentication is performed via chains of public-key certificates, and instead of storing certificates in centralized certificate repositories, they are stored and distributed by nodes themselves. The main idea of the proposed solution is the inclusion of a threshold scheme within the web-of-trust. During network initialization, nodes share the system's private-key, and each node holds one private-share. Instead of using private-keys for certificates signing, nodes use their private-shares. Each node in the network maintains a partial view of the web-of-trust, which is updated systematically through partial certificates exchanging protocol among neighboring nodes. The public-key authentication among nodes is performed via the combination of partial certificates chains. When a client node u needs to authenticate a public-key of another node v , both nodes merge and validate their partial views. The validation process is performed via the combination of all partial certificate signatures. If the

verification succeeds for a given node, all partial certificates issued for this node are marked as *trusted*. Otherwise, if the combination of signatures fails, they will be marked as *untrusted*. Finally, the node u tries to find a trusted certificates chain from node u toward node v . If such chain is found, the authentication is performed and then node u trusts the node v 's public-key.

Advantages and Drawbacks. The advantage of this scheme is that it is able to discover and isolate a high percentage of malicious nodes when compared to the previous schemes. The drawback is that the storage of partial certificates at each node, and their combination at each authentication, is both memory and time consuming.

6.2. *Reactive Models*

In this subcategory, the protocol of certificates collecting is executed on-demand when nodes need to verify a chain of certificates.

Funabiki et al.

Overview. Funabiki et al. [14] proposed a clustering-based trust model.

The certificates issuance are ensured by nodes themselves, and then stored at a particular node named CMN (Certificate Management Node) at each cluster. All the nodes of the cluster should request CMNs to collect the required certificates in order to verify the trust chain.

Advantages and Drawbacks. This scheme is interesting in the sense that users are detached of the certificates stocking. However, this system converges to the centralized models, while the certificates are stocked at particular nodes. Therefore, it involves many problems, such that the availability, central failure point, overhead related to replication of the certificates, etc.

ASNS - Kitada et al.

Overview. Kitada et al. [23, 25, 26] considered a web-of-trust model where every node has a repository that contains the node's certificates signed by some other nodes, and certificates delivered by the node itself for other nodes. Consider also source node and a destination node. The issue is how to find efficiently a chain of certificates from the source node to the destination node, and how to collect all the required certificates in the chain to carry out the necessary verifications? Kitada et al. explored this issue in mobile ad hoc networks and proposed the ASNS Protocol (Ad hoc Simultaneous Nodes Search). When a client node u wants to verify the certificate of a node v , it must firstly acquire a chain of public-key certificates from u to v . ASNS finds such a certificate chain as follows: u broadcasts the search request p to nodes that u directly trusts. If a node w receives the request p , w modifies the request p by adding its own certificate, and broadcasts it to the nodes that w directly trusts. If w is the destination node (v), it adds its own certificate to p and sends it to the source node (u). At the end of this search process, u receives p , which contains the chain of certificates from u to v . Then, it proceeds to signatures verification.

Advantages and Drawbacks. The major advantage of the scheme of Kitada et al. is that nodes are not required to interexchange public-key certificates. The protocol of certificates chain verification is performed in distributed way among nodes concerning. This scheme has the following problem. Since ASNS protocol uses broadcasting for trust chain discovery, the protocol will stop its iterations not when the destination node is discovered, but when all nodes in the network receive the request. Thus, ASNS suffers from a heavy communication cost because of broadcasting requests including certificates. Moreover, in the end of the protocol the source node will receive many chains of certificates, where finding one trust chain is sufficient for authentication.

Extensions and Improvements. Mohri et al. [41] proposed a modified approach of Kitada. They proposed to divide the process into two

phases: *certificates searching phase*, and *certificates collecting phase*. The first phase consists in executing the protocol proposed by Kitada et al. without adding the certificates in the request p . Instead, only identifiers are used. In the second phase, when the requester receives different candidate trust chains of identifiers, it chooses one chain. Then, it requests each concerned node to obtain certificate. Compared to the scheme of Kitada et al., this one achieves a low overhead. Moreover, in the second phase, choosing the certificates chain can be done based on performance or security criteria. For instance, the source may choose the shortest chain in order to optimize the signatures verifications process, or choose a chain containing nodes with a high degree of trust. Kambourakis et al. [24] addressed the same issue, and considered that the web-of-trust has the form of a binary tree. Hence, in order to respect the tree structure, each node in the network (1) is certified by only one of its neighboring nodes, and (2) it certifies, at the maximum, to two of its neighboring nodes. This scheme is efficient in terms of certificates chains recovery. However, this scheme may be unmanageable if the nodes in the network follow a high mobility, which generates a high complexity in order to maintain the tree structure.

6.3. *Modeling and Discussion*

In order to measure the degree of the possibility to authenticate a given node at a given moment, we have used SPN. The latter is adequate since the availability of a given node (or certificates repository server) at a given moment for a given node requester is probabilistic. In figure 13, we derived the SPNs corresponding to each trust model belonging to this category, and in table 6 the most used terminology in this subsection is defined.

6.3.1. *Successful Certification Probability Calculation (Capkun et al. Scheme)*

For instance in the model of Capkun et al. (cf. figure 13 (a)), at the arrival of an authentication request, the transition T_{AReq} will be reached, which

Table 6: Notations

Term	Description
λ_{AReq}	Interarrival duration rate of authentication requests.
λ_{ACC}	Interarrival duration rate of certification collaborators.
β	The number of required certification collaborators.
λ_{AHN}	Interarrival duration rate of helper nodes.
N_{hn}	Number of required helper nodes.
λ_{ACMN}	Interarrival duration rate of certificate management nodes.
N_{cmn}	Number of required certificate management nodes.
λ_{AIN}	Interarrival duration rate of intermediate nodes relating to a given certificates chain.
η	The length of a given certificates chain.
T_S	The transition representing the execution of the service of certificates chain verification.
$P\{T_S\}$	The probability of crossing the transition T_s in the SPN.
μ	The service duration rate of a given chain of certificates verification.

means that the request is ready to be executed, and thereby the next place will be configured by a token. The authentication process requires the availability of at least one chain of certificates between the source and destination. If such chain does not exist locally at the node, the latter should contact the helper nodes in order to collect more certificates. If a sufficient number N_{hn} of helper nodes are available, the transition T_{AHN} will be crossed N_{hn} times and then the next place will be set by N_{hn} tokens, which allows executing the authentication process, which is represented by the transition T_S . This probability can be calculated as follows [34]: $P\{T_S\} = \frac{\lambda_S}{\lambda_{AReq} + N_{hn}\lambda_{AHN} + \lambda_S} = \frac{\frac{1}{\mu}}{\lambda_{AReq} + N_{hn}\lambda_{AHN} + \frac{1}{\mu}} = \frac{1}{\mu\lambda_{AReq} + N_{hn}\mu\lambda_{AHN} + \frac{\mu}{\mu}} = \frac{1}{\mu\lambda_{AReq} + N_{hn}\mu\lambda_{AHN} + 1}$.

6.3.2. Overall Analysis

In table 7, we provide a comparison of the different anarchic trust models with respect to the rate of successful service of authentication $P\{T_S\}$. For each model, we provide the formula of $P\{T_S\}$ according to their respective SPNs depicted in figure 13, and present the management of certificates repositories and the way the certificates chains are recovered accordingly

(we note η , the length of certificates chains). Indeed, these two criteria influence the availability of certificate chains for verification and indirectly the rate of successful service of authentication $P\{T_S\}$. In this category, we remark that the probability of successful service of authentication is of the form: $P\{T_S\} = (\mu\lambda_{AReq} + \beta\mu\lambda_{ACC} + 1)^{-1}$, where λ_{ACC} is the interarrival duration rate of available Certification Collaborators: this is an abstraction of the required types of collaborators to achieve the collection of certificates. Depending on the model this would be equal to one of the following: λ_{AHN} , λ_{AIN} , or λ_{ACMN} . The metric β is a constant whose value depends on the trust model, which signify the number of required certification collaborators. The general form of the formula of $P\{T_S\}$ does not mean that the different trust models have all the same rate of successful service of authentication. Indeed, the latter depends on the values of λ_{ACC} (the way certification collaborators are managed), and the number of certification collaborators β . Depending on the considered trust model, the values of these two parameters belong to some different intervals dictated by the trust model nature and design. This allows further classifying the trust models of this category into three classes of performance (with respect to the rate of successful service of authentication) depending on the intervals in which evolve λ_{ACC} and β . The plot of $P\{T_S\}$ in figure 14 shows these classes.

Class A: In this class, we find in the top, the schemes of Capkun et al. [7, 8] and Ren et al. [48], which ensure a high level of certification service availability. Indeed, in this class, each node maintains a local repository, which is enriched, systematically, at each arrival of a neighboring node using the certificates exchanging protocol, which increases the service availability. Therefore, the collection of certificates is done locally at each authentication request. Indeed, when a requester node u needs to verify the node v 's public-key, they merge their local certificates repositories and find an appropriate certificates chain in the merged repository. If such chain is not found, u solicit some helper nodes in the neighborhood. This keeps the value of β reduced, and then makes this class not highly sensitive to the variation of β . We find, also in this class, the trust model of Omar et al. [43], but requiring more

Table 7: Comparison with respect to the rate of successful service of authentication $P\{T_S\}$

Scheme	Certificates repository	Chain recovery	$P\{T_S\}$	Quality of $P\{T_S\}$
Capkun et al. Ren et al. Omar et al.	Created, managed and interexchanged by nodes themselves.	The requester node, directly collects the certificates chain from its local repository.	$(\mu\lambda_{AReq} + N_{hm}\mu\lambda_{AHN} + 1)^{-1}$	High
Funabiki et al.	Managed by special nodes in the network (CMNs).	Request the CMN nodes.	$(\mu\lambda_{AReq} + N_{cmn}\mu\lambda_{ACMN} + 1)^{-1}$	Medium
Kitada et al. Mohri et al.	Only certificates issued by the node itself.	Broadcast the chain recovery request to all concerned nodes.	$(\mu\lambda_{AReq} + \eta\mu\lambda_{AIN} + 1)^{-1}$	Low
Kambourakis et al.	Only certificates issued by the node itself.	The chain recovery process follows the binary tree structure.	$(\mu\lambda_{AReq} + \eta\mu\lambda_{AIN} + 1)^{-1}$	Low

certificates, which is multiple of the threshold value k .

Class B: In this class, we find the scheme of Funabiki et al. [14], which ensures a medium level of certification service availability. This is due to the centralized management of certificates repository that are stored in special nodes (CMNs). This increases the value of $\lambda_{ACC} = \lambda_{ACMN}$. Since requester nodes require only the availability of CMNs, this class is not highly sensitive to the variation of β .

Class C: In this class, we find the trust models that ensure a low level of certification service availability. In the schemes of Kitada et al. [23, 25, 26] and Mohri et al. [41], there are no certificates repository maintained at each node. Instead, the chain of certificates is collected on-demand, at the moment of the authentication. Therefore, the success of the authentication depends strongly on the availability of the intermediate nodes in the chain of certificates. Hence, the requester node should use $\beta = \eta$ collaborators (intermediate nodes) in order to collect the chain of certificates. This makes this class strongly related to the certificates chain's length, contrary to the other classes. We find, also

in this class, the scheme of Kambourakis et al. [24], which use the same mechanism, and through the binary tree structure, they limit the maximal value of η at $\ln(m)$, where m is the network size.

6.4. *Overall Comparison*

In table 8, we give an overall comparison of trust models belonging to this category. We summarize the main lessons learnt from this comparison in the following points:

- The availability of the certification service depends on the ability of each node to collect any certificates chain relating to any other node in the network, in order to perform authentication processes. This property is related to the manner of managing the certificates repositories. If each node maintains a local repository, updated systematically through a protocol of certificates exchange with neighboring nodes, it can achieve a high level of availability since the retrieval of certificates is done locally at the node itself. If there is a central certificates repository in the network, the level of the service availability decreases.
- Like the first category, in this category also there is a computation overhead in terms of public-key computation. Another overhead of computation relating to this category, concerns the certificates chain verification cost. This problem concerns all trust models of this category. On the other hand, the most required resources are the storage and communication capacities. These overheads are induced by the storage and the exchange of certificates in the network, and make the differences among the solutions belonging to this category.
- The scalability of the certification service depends strongly on the number of certificates to store at each node. The certification service may be not scalable if the number of certificates is proportional to the network size. Another scalability issue may concern solutions that rely on a central certificates repository. Indeed, the latter may be

overloaded when solicited in a large scale, and even worse may be the single point of failure of the whole solution.

- In this category, the heterogeneity is handled through the web-of-trust mechanism, where nodes create themselves certificates using their own certification policy.

Table 8: Global comparison

Scheme	Availability	Resources Consumption	Scalability	Heterogeneity Support
Capkun et al. Ren et al. Omar et al.	High	Each node maintains a certificate repository, which incurs a high overhead.	No	Yes
Funabiki et al.	Medium	Each cluster supervisor node maintains all the certificates generated in the cluster, which incurs a high overhead.	No	Yes
Kitada et al.	Low	A high communication overhead due to requests broadcasting including certificates.	Yes	Yes
Mohri et al.	Low	Collecting certificates based on broadcasting, which involves a high communication overhead.	Yes	Yes
Kambourakis et al.	Low	High transmission and computation overheads in order to maintain the tree structure.	Yes	Yes

7. Conclusions

In this paper we focused on certification-based trust models in mobile ad hoc networks. We provided an overview of the objectives and requirements relating to certificates managements with respect to mobile ad hoc networks environments: service availability, resources awareness, scalability, and handling the heterogeneity. We have classified existing solutions into two approaches: (1) Authoritarian models, where the certification service is provided through one or several certification authorities. In order to take into consideration the above-mentioned requirements, and especially

availability and resources awareness, the certification service is distributed among a set of special nodes cooperation to provide the service through threshold cryptography. (2) Anarchic models, where each user in the network considers itself as a certification authority and establishes its own trust relationships according to some rules that may require the cooperation of other users in the network. Again, to take into consideration the above-mentioned requirements, some techniques are used to make certificates chain verification faster with low certificates storage overhead. We have further divided these two categories into fine grained sub-categories to illustrate the different organizational and performance aspects of the proposed solutions in the literature. We believe that the proposed taxonomy provides a global and precise insight over existing solutions, with a better understanding of the design choices decided by their authors. In table 9, we give an overall comparison of trust models analyzed through this paper.

In order to measure the service availability degree, we have modeled the reviewed certification-based trust models using SPNs (Stochastic Petri Nets), followed by comparisons and analytical discussions of each trust model. We have showed, in the authoritarian models, that there are two criteria that influence on the certification system availability. The first criterion is the coalition of servers providing the certification service: how to choose the servers? And how many servers can be available to respond to a certification requests? The second criterion is the choice of the threshold value (k). We have studied the impact of these two parameters on the successful certification rate of the existing trust models. This allowed us to further categorize the solutions into performance classes depending on the variation of these parameters dictated by the design of each trust model. In the other category of anarchic models, we have showed that there are two significant criteria that influence on the authentication service availability. The first criterion relates to the management of certificates repository servers, and especially their availability to respond to client nodes requests. The second criterion is the policy nature of certificates chains recovery, and especially, the induced length of certificates chains requiring verification during the certification process. We have then studied the impact of these parameters

on the rate of successful service of authentication. This culminated to the categorization of existing solutions into performance classes depending on the design of each trust model.

This survey should help shed some light on certification-based trust models in mobile ad hoc networks. It should be especially useful to get a global and precise insight of existing solutions through a fine grained taxonomy and a thorough performance modeling, evaluation and comparison.

Table 9: Overall comparison with respect to: trust model architecture (authoritarian/anarchic), distribution (fully/partially), with/without mechanism of revocation, static/dynamic threshold value.

Scheme	Authoritarian Model	Anarchic Model	Fully Distributed	With Revocation Mechanism	Dynamic Certification Threshold
Zhou and Haas	✓	×	×	×	×
MOCA	✓	×	×	✓	×
Dong et al.	✓	×	×	×	×
Kong et al.	✓	×	✓	✓	×
Raghani et al.	✓	×	✓	×	✓
DICTATE	✓	×	×	✓	×
NetTRUST	✓	×	×	×	×
Ge et al.	✓	×	×	✓	×
Seys and Preneel	✓	×	×	✓	×
Wang et al.	✓	×	×	×	×
Xu and Iftode	✓	×	×	✓	×
Capkun et al.	×	✓	✓	✓	–
Ren et al.	×	✓	✓	×	–
Omar et al.	×	✓	✓	×	×
Funabiki et al.	×	✓	✓	✓	–
Kitada et al.	×	✓	✓	✓	–
Mohri et al.	×	✓	✓	✓	–
Kambourakis et al.	×	✓	✓	×	–

8. Bibliography

- [1] Abdulrahman A. The PGP Trust Model. *The Journal of Electronic Commerce*; 1997.
- [2] Ayday E, Fekri F. A protocol for data availability in Mobile Ad-Hoc Networks in the presence of insider attacks. *Ad Hoc Networks*; 2010.
- [3] Aziz B, Nourdine E, Mohamed E-K. A Recent Survey on Key Management Schemes in MANET. In *proceedings of Information and Communication Technologies*; 2008.
- [4] Buchegger S, Le-Boudec J-Y. Performance analysis of the CONFIDANT protocol. In *Proceedings of 3rd ACM International Symposium, on Mobile Ad Hoc Networking and Computing*; 2002.
- [5] Bonh D, Franklin M. Identity-Based Encryption from Weil Pairing. *Advances in Cryptology (Springer Verlag)*; 2001.
- [6] Boukerch A, Xu L, EL-Khati K. Trust based security for wireless ad hoc and sensor networks. *Computer Communications*; 2007.
- [7] Capkun S, Buttyan L, Hubaux J. Small Worlds in Security Systems - an Analysis of the PGP Certificate Graph. In *Proceedings of New Security Paradigms Workshop (ACM)*; 2002.

- [8] Capkun S, Buttyan L, Hubaux J. Self-organized Public Key Management for Mobile Ad hoc Networks. *IEEE Transactions on Mobile Computing*; 2003.
- [9] Corson S, Macker J. Mobile ad hoc networking - Routing Protocol Performance Issues and Evaluation Considerations. *IETF RFC 2501*; 1999.
- [10] Calafate C-T, Oliver J, Cano J-C, Manzoni P, Malumbres M-P. A distributed admission control system for MANET environments supporting multipath routing protocols. *Microprocessors and Microsystems*; 2007.
- [11] Deng H, Li W, Agrawal D. Routing Security in Wireless Ad hoc Networks. *IEEE Communications Magazine*; 2002.
- [12] Douceur J. The Sybil Attack. In *Proceedings of the International Workshop of Peer-to-Peer Systems*; 2002.
- [13] Dong Y, Sui A, Yiu S, Li V, Hui L. Providing Distributed Certificate Authority Service in Cluster-based Mobile Ad hoc Networks. Elsevier, *Computer Communications*; 2007.
- [14] Funabiki S, Isohara T, Kitada Y, Takemori K, Sasase I. Public Key Management Scheme with Certificate Management Node for Wireless Ad Hoc Networks. In *Proceedings of the International Multiconference on Computer Science and Information Technology*; 2006.
- [15] Giordano S. Mobile Ad-Hoc Networks. *Handbook of Wireless Networks and Mobile Computing*, John Wiley and Sons; 2001.
- [16] Ge M, Lam K-Y, Gollmann D, Chung S-L, Chang C-C, Li J-B. A robust certification service for highly dynamic MANET in emergency tasks. *Wiley InterScience: International Journal of Communication Systems*; 2009.
- [17] Haas P-J. *Stochastic Petri Nets: Modelling, Stability, Simulation*. Springer Series in Operations Research; 2002.

- [18] Haas P-J. Estimation of delays in non-regenerative stochastic petri nets. Report; 2007.
- [19] Herzberg A, Jarecki S, Krawczyk H, Yung M. Proactive secret sharing or: How to cope with perpetual leakage. In Proceedings of Crypto'95 LNCS (Springer-Verlag); 1995.
- [20] Hwang M, Lu E, Lin I-C. A practical (t, n) threshold proxy signature scheme based on the RSA cryptosystem. IEEE Transactions on Knowledge and Data Engineering; 2003.
- [21] He Q, Wu D, Khosla P. SORI: a secure and objective reputation-based incentive scheme for ad-hoc networks. In Proceedings of IEEE WCNC'04; 2004.
- [22] Janzadeh H, Fayazbakhsh K, Dehghan M, Fallah M-S. A secure credit-based cooperation stimulating mechanism for MANETs using hash chains. Future Generation Computer Systems; 2009.
- [23] Kitada Y, Arakawa Y, Takemori K, Watanabe A, Sasase I. On demand distributed public key management using routing information for wireless ad hoc networks. IEICE Transactions on Information and Systems; 2005.
- [24] Kambourakis G, Konstantinou E, Douma A, Anagnostopoulos M, Fotiadis G. Efficient Certification Path Discovery for MANET. EURASIP Journal on Wireless Communications and Networking; 2010.
- [25] Kitada Y, Watanabe A, Takemori K, Sasase I. On demand distributed public key management for wireless ad hoc networks. IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PacRim); 2005.
- [26] Kitada Y, Watanabe A, Takemori K, Sasase I. On demand distributed public key management without considering routing tables for wireless ad hoc networks. Asia Pacific Symposium on Information Technology (APSITT); 2005.

- [27] Kong J, Zerfos P, Luo H, Lu S, Zhang L. Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks. In Proceedings of International Conference on Network Protocols, IEEE Computer Society; 2001.
- [28] Lauter K. The advantages of Elliptic Curve Cryptography For Wireless Security. IEEE Wireless Communications; 2004.
- [29] Luo J, Hubaux J, Eugster P. DICTATE - Distributed Certification Authority with Probabilistic Freshness for Ad hoc Networks. IEEE Transactions on Dependable and Secure Computing; 2005.
- [30] Luo H, Lu S. Ubiquitous and Robust Authentication Services for Ad hoc Wireless Networks. Technical Report, UCLA Computer Science; 2000.
- [31] Liu Y, Lia K. A novel reputation computation model based on subjective logic for mobile ad hoc networks. Future Generation Computer Systems; 2010.
- [32] Luo J, Liu X, Fana M. A trust model based on fuzzy recommendation for mobile ad-hoc networks. Computer networks; 2009.
- [33] Luo H, Zerfos P, Kong J, Lu S, Zhang L. Self-securing ad hoc wireless networks. In Proceedings of the Seventh International Symposium on Computers and Communications (ISCC'02); 2002.
- [34] Marsan M-A. Stochastic Petri Nets: An Elementary Introduction. Report; 1988.
- [35] Marchanga N, Dattab R. Collaborative techniques for intrusion detection in mobile ad-hoc networks. Ad Hoc Networks; 2008.
- [36] Merwe J-V-D, Dawoud D, McDonald S. A Survey on Peer-to-Peer Key Management for Mobile Ad Hoc Networks. ACM Computing Surveys; 2007.
- [37] Marias G-F, Georgiadis P, Flitzanis D, Mandalas K. Cooperation enforcement schemes for MANETs: A survey. In wireless communication and mobile computing; 2006.

- [38] Michiardi P, Molva R. CORE: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In Proceedings of 6th IFIP Communication and Multimedia Security Conference; 2002.
- [39] Mishra A, Nadkarni K-M. Security in wireless ad hoc networks - A Survey. In The Handbook of Ad Hoc Wireless Networks, M. Ilyas, Ed. Boca Raton: CRC Press; 2002.
- [40] Menezes A, Van-Oorschot P, Vanstone S. The Handbook of Applied Cryptography. CRC Press; 1996.
- [41] Mohri H, Yasuda I, Takata Y, Seki H. Certificate Chain Discovery in Web of Trust for Ad Hoc Networks. In Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07); 2007.
- [42] Omar M, Challal Y, Bouabdallah A. NetTRUST: mixed NETworks Trust infrastrUcture baSed on Threshold cryptography. In Proceedings of Securecom'07/SECOVAL (IEEE); 2007.
- [43] Omar M, Challal Y, Bouabdallah A. Reliable and fully distributed trust model for mobile ad hoc networks. Computers and Security; 2009.
- [44] Paterson K-G. ID-based Signatures from Pairing on Elliptic Curves. Cryptology ePrint Archive, Report; 2004.
- [45] Perkins C. Ad Hoc Networking. Addison-Wesley Professional; 2000.
- [46] Plesse T, Adjih C, Minet P, Laouiti A, Plakoo A, Badel M, Muhlethaler P, Jacquet P, Lecomte J. OLSR performance measurement in a military mobile ad hoc network. Ad Hoc Networks; 2005.
- [47] Papadimitratos P, Haas Z. Securing Mobile Ad Hoc Networks. In The Handbook of Ad Hoc Wireless Networks, M. Ilyas, Ed. Boca Raton: CRC Press; 2002.
- [48] Ren K, Li T, Wan Z, Bao F, Deng R, Kim K. Highly Reliable Trust Establishment Scheme in Ad hoc Networks. Elsevier, Computer Networks; 2004.

- [49] Raghani S, Toshniwal D, Joshi R. Dynamic Support for Distributed Certification Authority in Mobile Ad Hoc Networks. In Proceedings International Conference on Hybrid Information Technology (IEEE); 2006.
- [50] Shamir A. How to Share a Secret. Communication of the ACM; 1979.
- [51] Shirey R. Internet Security Glossary. RFC 2828; 2000.
- [52] Seys S, Preneel B. Authenticated and Efficient Key Management for Wireless Ad Hoc Networks. In Proceedings of the 24th Symposium on Information Theory in the Benelux, Werkgemeenschap voor Informatie-en Communicatietheorie; 2003.
- [53] Satizabal C, Hernandez-Serrano J, Pegueroles J. Building a virtual hierarchy to simplify certification path discovery in mobile ad-hoc networks. Wired/Wireless Internet Communications; 2007.
- [54] Vimala N, Balasubramaniam R. Distributed Key Management Scheme for Mobile Ad-Hoc Network-A Survey. Global Journal of Computer Science and Technology; 2010.
- [55] Wang W, Zhu Y, Li B. Self-managed Heterogeneous Certification in Mobile Ad hoc Networks. In the Proceedings of the Vehicular Technology Conference (IEEE); 2003.
- [56] Wu S-L, Tseng Y-C. Wireless Ad Hoc Networking: Personal-Area, Local-Area, and the Sensory-Area Networks. CRC Press; 2007
- [57] Xu G, Iftode L. Locality driven key management architecture for mobile ad-hoc networks. In Proceedings of the First IEEE International Conference on Mobile and Sensor Networks (MASS'04); 2004.
- [58] Yu J-Y, Chong P-H-J. A survey of clustering schemes for mobile ad hoc networks. Communications Surveys and Tutorials; 2005.
- [59] Yi S, Kravets R. MOCA - Mobile Certificate Authority for Wireless Ad hoc Networks. In Proceedings of the Second Annual PKI Research Workshop; 2003.

- [60] Yick J, Mukherjee B, Ghosal D. Wireless sensor network survey. Computer Networks; 2008.
- [61] Zhou L, Haas Z. Securing Ad hoc Networks. IEEE Networks; 1999.
- [62] Zimmermann P. The Official PGP User's Guide. MIT Press; 1995.
- [63] Zhang Y, Lee W. Intrusion Detection in Wireless Ad hoc Networks. In Proceedings of MobiCom'00 the sixth Annual of the International Conference on Mobile Computing and Networking; 2000.
- [64] Zouridaki C, Mark B-L, Hejmo M, Thomas R-K. E-Hermes: A robust cooperative trust establishment scheme for mobile ad hoc networks. Ad Hoc Networks; 2009.

Figure 11: Plots of $P\{T_S\}$ in function of k and λ_{AS} with $\mu = 1/20$ and $\lambda_{Req} = 10$. The authoritarian models are classed in four categories on the plot according to the interval of variation of both k and λ_{AS} of each model. The impact of k depends on the way its value is maintained: static/dynamic. In the case of dynamic threshold k , the latter is systematically adjusted in order to keep high the availability of the service of certification, and hence do not much influence the probability $P\{T_S\}$ (Class A). In the case of static threshold k , the availability of the service of certification $P\{T_S\}$ depends mainly on the availability degree of nodes who deliver partial certificates (the parameter λ_{AS}). The interval of variation of λ_{AS} depends on the availability of servers. The schemes where all nodes are servers or auxiliary servers can be systematically integrated are classified under (Class B). The schemes where the number of servers is limited are classified under (Class C). The schemes where the conditions to execute the service of certification are complex, are classified under (Class D).

local



local

Figure 12: Capkun et al.

(a)

eta

Figure 13: SPNs of anarchic models

Figure 14: Plots of $P\{T_S\}$ in function of β and λ_{ACC} with $\mu = 1/30$ and $\lambda_{AReq} = 10$. The anarchic models are classed in three categories on the plot according to the interval of variation of the degree of availability of collaborator nodes λ_{ACC} and their number β . The interval of variation of λ_{ACC} and β depends on the way to collect the certificates chains: in proactive way (Class A), in reactive way with low/high number of collaborator nodes (Class B/Class C).