STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **16**, December 2026, pp 6476–6493.
Published online in International Academic Press (www.IAPress.org)

RSA–Based Distributed Certificate Authorities in MANETs Design Requirements and Evaluation Framework

Ahmad K. Alomari [1,*], Emad A. Az-Zo'bi [2]

[1]*Department of Computer Science, Faculty of Science and Information Technology, Jadara University, Irbid, Jordan*
[2]*Department of Mathematics and Statistics, Mutah University, Mutah P.O.Box 7, Al Karak 61710, Jordan*

Abstract Mobile Ad Hoc Networks (MANETs) have emerged as an important communication paradigm due to their ability to establish wireless connections without relying on fixed infrastructure. The decentralized and dynamic nature of MANETs makes them suitable for military operations, disaster recovery, emergency communications, and remote-area networking. However, the absence of centralized administration introduces significant security challenges, particularly in node authentication, certificate management, and key distribution. Traditional authentication mechanisms based on centralized Certificate Authorities (CAs) are not well suited for MANET environments because they create a single point of failure and may become inaccessible due to network partitioning or node mobility. To address these limitations, several distributed certificate authority schemes have been proposed in the literature. These approaches distribute the CA functionality among multiple nodes to improve availability and fault tolerance. Nevertheless, many existing solutions rely on predefined trusted nodes, static trust assumptions, or partially centralized architectures, which reduce their effectiveness in highly dynamic environments. Furthermore, certificate renewal and revocation processes often generate additional communication overhead and increase management complexity. In this work, a fully self-organized certificate authority model based on the RSA cryptographic algorithm is considered. Every node in the network acts as a dealer and participates in distributed key management. The proposed framework aims to reduce dependence on dedicated authority nodes. In addition, polynomial–based secret sharing techniques are employed to support certificate renewal and revocation procedures. The motivation is to coordinate distributed trust mobility while making the cryptographic setup and certificate–management assumptions explicit. The reported performance indicators are retained as descriptive results; their reproducibility and statistical uncertainty require further documentation. A consistent binding between the shared secret and a common RSA signing key remains a technical requirement to be resolved.

Keywords MANETs, Secret Sharing, Threshold RSA, Certificate Authority, Certificate Revocation.

AMS 2010 subject classifications 94A60, 94A62, 94A05, 68M10

DOI: 10.19139/soic-2310-5070-4088

1. Introduction

Over the past decades, the use of mobile digital devices, like laptops, tablets, smart phones or personal digital assistants (PDAs), has grown rapidly. Wireless interconnection supports access to information across devices and locations. This motivates interest in infrastructure–independent communication and in secure routing for mobile nodes [1, 2]. One important wireless network architecture is the Mobile Ad Hoc Network (MANET).

A MANET consists of a network of wireless mobile nodes. Their main characteristic is that they establish dynamic, temporary and changing network topologies without a fixed infrastructure. People, vehicles or other communication devices can benefit from internet access in remote areas where the infrastructure is missing or is

*Correspondence to: Ahmad Alomari (Email: a.alomari2@jadara.edu.jo). Department of Computer Science, Faculty of Science and Information Technology, Jadara University, Irbid, Jordan.

ISSN 2310-5070 (online) ISSN 2311-004X (print)
Copyright © 2026 International Academic Press

poor, or when a wireless extension is needed to gain admission to that infrastructure [3]. In the mobile ad hoc network, the communication is accomplished in two ways: either directly, when a node communicates with any node in its radio range; or with the help of intermediate node(s), for the nodes that are out of range. All the nodes that establish communication in both of these situations group to form a wireless network, also regarded as a mobile ad hoc network [4, 5, 6].

At the initialization of the ad hoc network, the groups of nodes that form it have not had any previous connection with each other and did not exchange any secret information. Ad hoc networks use key management protocols in an attempt to secure themselves. So, in order to create these key management protocols, a secure communication infrastructure has to be set, a task that is very challenging due to the lack of a fixed server or any other kind of trusted authority. Generally, key management is grouped into three categories, namely key agreement, key pre-distribution and key distribution, all with their own restriction when applied to an ad hoc network.

MANETs often use several certificate authorities (CAs), dispersed throughout the network, each with a regularly updated portion of the secret key, due to the lack of centralized administration and an existing infrastructure. The RSA key management system we provide in this work is based on a fully distributed certificate authority model.

The existing authentication services are based on centralized management approaches as either key distribution centers or CAs. The centralized approach is considered to be acceptable in cases where a specific node can be protected and it can be reached by other nodes of the network. However, because a centralized method will experience a single point of service denial and may be unavailable to network nodes that need CA services, it is insufficient for wireless ad hoc networks. Therefore, a stronger CA strategy has to be applied.

Typically, a trustworthy CA is employed to verify the public keys' dependability. In order to issue a digital certificate and connect the public key (which is part of the digital certificate) to that specific node, the CA is asked to verify each public key before the node distributes it to the designated parties. The CA then uses its own private key to sign the digital certificate. A PKI (public key infrastructure) aided by a trustworthy CA seems to be the most promising approach for protecting MANETs.

Recent studies address secure routing, routing optimization, and authentication in MANETs [7, 8, 9, 10, 11]. These studies provide network–security context; they should not all be characterized as blockchain–based or distributed CA constructions. Routing performance and certificate-management performance are related but distinct evaluation objectives.

The motivation of this work is to reduce dependence on a continuously reachable authority while supporting certificate renewal and revocation in a changing MANET. The intended setting is a provisioned group, such as an emergency-response team, whose authenticated members subsequently experience mobility and intermittent connectivity. The proposed all-node dealer organization is examined as a certificate-management design; combining RSA with secret sharing alone is not claimed as a new cryptographic primitive.

The paper focuses on organizing an RSA-based, fully distributed CA around node participation in sharing, renewal and revocation, identifying the common-parameter, verification and completion requirements needed to make that organization precise, and reporting comparative performance indicators and correlations while distinguishing descriptive observations from claims requiring reproducible experiments. The all-node architecture is retained, but its advantages over other CA architectures remain conditional on a fair comparison and a complete cryptographic instantiation.

2. Preliminaries

This section includes different models of completely distributed certificate authorities, along with a brief explanation of the RSA cryptographic method and the Shamir secret sharing mechanism.

2.1. Shamir Secret Sharing Scheme

Shamir [12] introduced the threshold secret sharing scheme using polynomial interpolation. Each participant receives a share of the secret. Any t distinct valid shares suffice to reconstruct the secret; all n shares are not required. Fewer than t shares reveal no information about a uniformly sampled secret under the standard scheme.

Secret reconstruction is distinct from threshold signing, which should avoid disclosing the CA private key. Therefor, we state that this type of arrangement involves one parameter and n participants. The dealer reveals the secret to the players only under specific conditions: the secret can be reconstructed by integrating the shares held by any group of t participants or more, where t represents the threshold value, whereas any group comprising fewer than t players is unable to do so. We refer (n, t) threshold scheme to this system. It is simple to demonstrate the security of Shamir's scheme: in an (n, t) scheme, it can be demonstrated that it makes no difference if an attacker gets $t-1$ legitimate shares to work with or none at all; unless he obtains t shares or more, guessing is the best way to discover the secret.

There are situations when a secret must be disclosed to (at least) users without any user being able to figure it out on their own. Shamir reveals the problem in his work and offers a secret sharing plan that uses polynomial interpolation as a recovery method. Therefore, each user has a pair $(x_i, f(x_i))$, where $x_i \neq 0$, where $f(x)$ is a polynomial of degree $t-1$, and the secret is given by $f(0)$. In this arrangement, one needs at least t shares to recover f, then $f(0)$. With these parameters, to share a secret a_0 into d shares, one needs to choose $t-1$ random numbers $(a_{t-1}, \ldots, a_1)$ to construct the polynomial.

$$f(x) = a_0 + a_1 x + \cdots + a_{t-1} x^{t-1}.$$

After that, it shares the sub secret between the n users, where $S_i = f(i)$, where $i = 1, \ldots, n$ and securely transfers the share S_i to the users $i = 1, \ldots, n$.

To reconstruct the initial secret S, we need a subgroup of at least t users to exchange the sub-secrets between them. Next, each user of t group will get t distinct points (i, S_i) of the polynomial. Afterwards the Lagrange interpolation is used to calculate the coefficient of the polynomial

$$f(x) = \sum_{i=1}^{t} S_i \prod_{\substack{j=1 \\ j \neq i}}^{t} \frac{x-j}{i-j},$$

where $f(0) = a_0 = S$. This means that the shared secret may be expressed as:

$$S = \sum_{i=1}^{t} S_i \prod_{\substack{j=1 \\ j \neq i}}^{t} \frac{-j}{i-j}.$$

Shamir interpolation schemes assume that all shares are evaluations of a polynomial of degree at most $t-1$ over the same finite field, at distinct, non-zero participant identifiers. The non-constant coefficients are sampled independently and uniformly from this field. These schemes alone do not define an RSA threshold construction; in particular, shares computed using independently chosen RSA parameters cannot be combined via standard Shamir interpolation without additional justification.

2.2. RSA Cryptographic Algorithm

RSA systems, created by Ronald Rivest, Adi Shamir, and Len Adleman at MIT in 1978 [13], are the inaugural reversible public key cryptosystems, designed for both encryption and digital signature applications.

For two distinct primes P and Q, the RSA modulus is $N = PQ$ and $\phi(N) = (P-1)(Q-1)$. The public exponent e satisfies $1 < e < \phi(N)$ and $\gcd(e, \phi(N)) = 1$. The private exponent d satisfies $ed \equiv 1 \pmod{\phi(N)}$. The public key is (N, e); d and the factorization remain secret. The uppercase N denotes the modulus, whereas n denotes the number of participants [14].

The textbook operations are $c = m^e \bmod N$ and $m = c^d \bmod N$ for $0 \leq m < N$. A signature applies the private-key operation to an encoded message representative, not directly to unpadded text. Signature verification checks the recovered encoding according to the selected signature scheme [14].

2.3. Fully Distributed Models

In Ubiquitous and Robust Authentication Services for Ad Hoc Wireless Networks [15], Luo and Lu introduced us to the Fully Distributed Certificate Authority Approach. A (n, k) threshold distribution strategy is used to ensure that each node receives its share at the time of network entry. Additionally, it employs proactive and verifiable secret sharing techniques to protect the network against denial of service attacks and prevent the compromise of the certificate signing key.

It is considered that nodes must watch the behavior of their neighbor's in order to build trust. The nodes are also required to maintain their own certificate revocation list (CRL). A rogue node's certificate is added to the CRL of the node that found it, and a "accusation" is sent throughout the network. All of the other nodes mark the accused node as a suspect, making it suspicious. The accusatory certificate must be withdrawn in order for the allegation to be ignored.

Dong et al. [16] studied distributed CA services in cluster–based MANETs. Their architecture addresses locating sufficient CA servers and proactive secret-share updates. These are relevant comparison dimensions for the present all–node organization: service discovery, update traffic and availability of enough shareholders. The comparison concerns architecture; it does not establish that the present proposal has lower overhead or better availability.

Crites et al. [17] introduced the first Threshold Structure–Preserving Signature (TSPS) scheme designed for decentralized environments where multiple parties jointly generate a signature without relying on a central authority. The authors addressed the limitations of existing threshold signature and structure–preserving signature schemes, particularly their inability to provide non–interactive threshold signing while maintaining compact signatures and efficient verification. The proposed framework is built upon indexed Diffie–Hellman message spaces and extends conventional structure-preserving signatures to support threshold operations. The scheme enables signers to independently produce partial signatures that can later be aggregated into a single valid signature without requiring communication among participants. Security analysis demonstrated that the proposed construction achieves existential impossibility of forgery under chosen indexed–message attacks in the random oracle model. Furthermore, the study showed that the scheme can serve as an efficient replacement for threshold issuance anonymous credential systems by reducing computational overhead while preserving strong security guarantees.

To comply with the mobility of the nodes and the continuous changes in membership, a fully self–organized trust establishment method is embraced. These methods attempt to address the issue of key management in MANETs. These approaches do, however, have certain drawbacks, such as node reliance, administrator availability, and congestion. They provide three potential strategies to address the issue of key management.

Distributing PKI's functionalities to every network node is the initial step. However, because of the network's dynamics, it is challenging to guarantee that every member will be available. The second option is to make a specific group of nodes permanent PKI members, allowing them to roam freely throughout the network space. As will be explained later, the clustered architecture that forms the basis of the final solution is where the cluster–heads make up the PKI members. They do a comparison analysis of the second and final solutions in their job. Their methodology will be explained in the sections that follow. Their strategy, which draws inspiration from the Threshold Secret Sharing Scheme, makes use of both the clustering technique and the partially distributed PKI solution. They would reveal their plan to combine the functions of several cluster leaders into one service known as the Council. Every Council node will act equally and carry out the cluster head functions using the (n, k) threshold approach. This Council's primary duty will be key management. At least k nodes out of n Council members must participate for a certificate to be verified. Now, even if several cluster heads (but no more than $min\{k, n - k + 1\}$) are compromised, the key management cluster head function will continue to function. They suggested a brand–new design, which they refer to as "Council"-based clusters. In order to function as Council–based clusters across the network, the concept employs a collaborative method, which maximizes efficiency. Following the formation of the Council–based clusters, each Council member may implement the (n, k) threshold scheme such that, in order to carry out any CA function, at least k cluster heads out of n must collaborate.

Yan et al. [18] proposed a common share authentication method for image secret sharing (ISS) that supports both dealer–participatory and dealer–non–participatory authentication. The study addressed a significant limitation of previous ISS schemes, where authentication often required the dealer's involvement or introduced additional

computational complexity and pixel expansion. To overcome these challenges, the authors combined the advantages of polynomial–based image secret sharing and visual secret sharing (VSS) into a unified framework. The proposed scheme uses a screening operation to integrate authentication information within the generated shares while maintaining lossless secret image recovery and avoiding pixel expansion. Unlike traditional approaches that rely on information hiding or auxiliary encryption techniques, the method enables share authentication through simple XOR and stacking operations, reducing computational overhead. The scheme also allows participants to verify the authenticity of shares before reconstruction, helping to identify fake shares and prevent unsuccessful recovery processes. Experimental results demonstrated that the proposed method preserves image quality, achieves lossless reconstruction of the secret image, and provides effective authentication under different threshold settings. Furthermore, the scheme supports secure and efficient image sharing while maintaining the essential properties of threshold–based secret sharing systems, making it suitable for practical multimedia security applications. This image–sharing work [18] concerns authentication of image shares; it is included as background on share validation and is not a threshold–RSA construction or a direct MANET CA benchmark.

Alomari [19] proposed a fully distributed certificate authority for Mobile Ad Hoc Networks (MANETs) based on polynomial secret sharing and elliptic curve cryptography. The proposed framework distributes the certificate authority's private key among multiple nodes, eliminating the need for a centralized trusted entity. Nodes collaboratively generate and validate certificates, improving authentication reliability in dynamic network environments. The use of elliptic curve cryptography reduces computational overhead while maintaining a high level of security. The scheme is resistant to single-node compromise because an attacker must obtain multiple key shares to reconstruct the secret key. Experimental results demonstrated improved efficiency and secure certificate management. The study concluded that the proposed approach provides a scalable and secure authentication solution for MANETs.

Kamal and Iwamura proposed an improved secure multi–party multiplication scheme for threshold secret sharing systems [20]. The study addressed the problem of polynomial degree expansion during multiplication, which typically increases the number of shares and servers required for computation. The proposed method enables secure multiplication using only $N = k$ servers while maintaining a (k, n) threshold secret sharing structure. By allowing each server to hold multiple shares and applying a recombination vector technique, the scheme reduces communication and reconstruction overhead. The authors also provided a security analysis demonstrating resistance against semi–honest adversaries. Experimental evaluation showed improvements in efficiency compared with existing methods such as the Watanabe and TUS schemes. The study concluded that the proposed approach offers a secure and practical solution for secure multi–party computation in distributed environments.

The related studies address different parts of network security, including secure computation [20], authentication [21], and routing [22] do not by themselves establish that an online distributed CA is insecure or that the proposed service is independent of mobility. In an all–node CA, mobility affects access to enough participants for certification sufficiently current revocation information.

The localized certificate-service approach of Luo and Lu [15] and the elliptic–curve CA design [19] are relevant architectural precedents. The present RSA–based proposal must specify its enrollment assumptions, compromise threshold, signing–share verification, setup cost and partition policy before a direct advantage can be established. A new threshold primitive is not proposed. Shoup [23], Damgård and Koprowski [24] provide threshold–RSA foundations with specific assumptions; selecting one requires an explicit implementation and security model.

Sbayti and Housni [25] evaluate ACO–based OLSR routing in a vehicular ad hoc network using SUMO and NS3. Their throughput, packet–delivery, delay and routing–overhead metrics are relevant to network–level evaluation, but routing gains cannot be attributed to certificate management. A fair CA comparison should hold routing, traffic and mobility settings constant across alternatives.

Rahman et al. [26] examine stacked-ensemble intrusion detection using NSL–KDD. This study informs the distinction between detection and revocation: a detection output is evidence to assess, not sufficient authorization to revoke a certificate. Detection errors, false revocations and propagation delay should therefore be reported separately.

3. Fully Self-Organised Certificate Authority using RSA Algorithm in MANETs

The revised design considers an authenticated roster of n participants during one configuration epoch, with $2 \leq t \leq n$ as the reconstruction threshold. Each participant has a distinct nonzero field identifier x_j. Enrollment authenticates node identities and binds the epoch, roster and CA public key; an unauthenticated Diffie–Hellman exchange alone is insufficient. No assumption that every node has $t-1$ honest one-hop neighbors is made. Certificate service instead requires reachability of enough eligible shareholders, possibly over multiple hops. Local monitoring supplies evidence of suspected misbehavior, not automatic revocation.

The CA private key is represented by distinct signing shares held by the participants; a complete private key is not issued to every node. The threshold t specifies how many valid shares are required by the selected construction. The CA public key must be authenticated after successful key generation and before certificates are accepted.

For Shamir sharing of a field secret, a polynomial of degree at most $t-1$ gives one share to each participant. Any t distinct valid shares reconstruct the secret, whereas fewer than t shares reveal no information when the non–constant coefficients are sampled independently and uniformly. This information-theoretic statement applies to field sharing.

The revised specification separates the prime-field sharing component from RSA signing. Authors in [24] provide a published route to threshold RSA without a trusted dealer, with construction-specific integer sharing, verification and adversary assumptions. That route is identified as a candidate for implementation; it is not interchangeable with the field equations below. All RSA signing shares must be generated for the same public key (N, e).

3.1. Initialization of Scheme

Each enrolled node may contribute as a dealer in the sharing layer. The description below defines consistent sub–share notation and verification over a common prime field. It clarifies the original distribution diagram without claiming that this field–sharing procedure generates an RSA key.

For one setup epoch, participants agree on a roster, session identifier and parameter digest. Each message binds the epoch, dealer, recipient and message type. A participant checks identity, freshness and commitment consistency before accepting a sub–share. Setup completes only with a common qualified–dealer set Q and an agreed transcript digest. Missing or disputed contributions cause an abort or a restart under a newly agreed roster; a node must not silently use a different dealer set. Membership changes require a new authenticated configuration. This completion rule specifies the safety condition; a concrete agreement protocol and timeout values remain implementation inputs.

Let $\ell > n$ be prime and let G be a cyclic group of order ℓ with nonidentity generators g and h. No participant may know $\log_g(h)$. All dealers use the same G, ℓ, g, h and identifiers $x_j \in \mathbb{F}_\ell \setminus \{0\}$. The group and the RSA modulus are separate parameters [27].

Dealer D_i samples coefficient vectors $(a_{i0}, \ldots, a_{i(t-1)})$ and $(b_{i0}, \ldots, b_{i(t-1)})$ over $\mathbb{F}_\ell$. Its secret contribution is a_{i0}; the b coefficients provide independent blinding. Nonconstant coefficients are sampled uniformly

$$f_i(X) = \sum_{k=0}^{t-1} a_{ik} X^k, \qquad r_i(X) = \sum_{k=0}^{t-1} b_{ik} X^k.$$

All polynomial arithmetic is in $\mathbb{F}_\ell$.

The dealer publishes commitments $C_{ik} = g^{a_{ik}} h^{b_{ik}}$ for $k = 0, \ldots, t-1$. The unknown logarithm relation between g and h supports computational binding; the blinding coefficients hide the committed values. A setup that lets a dealer choose $h = g^\alpha$ while retaining α does not satisfy the unknown-relation requirement [27].

The private sub-share pair for recipient D_j is $s_{ij} = f_i(x_j)$ and $r_{ij} = r_i(x_j)$. In this notation i always identifies the dealer and j the recipient. The threshold remains t; r_{ij} is a blinding share, not a threshold. Recipient D_j accepts the pair only if

$$g^{s_{ij}} h^{r_{ij}} = \prod_{k=0}^{t-1} C_{ik}^{x_j^k}.$$

Exponents are evaluated modulo ℓ and group operations in G. An invalid or missing pair is reported through an authenticated complaint bound to the session. The dealer delivers each remote pair though an authenticated confidential channel and retains its own local pair. Commitments are distributed authentically to all participants with common transcript required to prevent inconsistent views. Figure 1 separates these private and public flows.

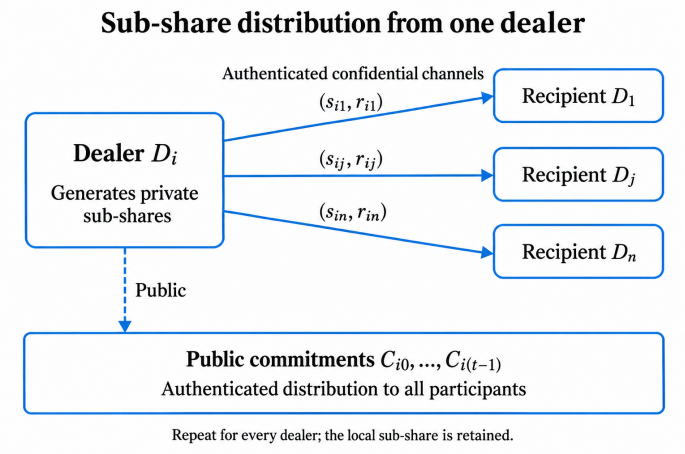


Figure 1. Private sub–share delivery and public commitment distribution from a representative dealer D_i. The recipient index j differs from i on a remote channel; the dealer retains its local pair. The pattern repeats for every dealer.

Selecting the largest modulus cannot convert independently generated shares into shares of one RSA key. The corrected aggregation below concerns only a common–field secret. An RSA implementation must instead use the complete key–generation and signing algorithms of a compatible threshold–RSA construction [24].

After complaint handling and agreement on Q, each recipient aggregates only accepted contributions from that same dealer set. Disputed shares must not be published ad hoc: disclosure and exclusion rules must follow the chosen verifiable–sharing protocol, so complaint handling does not leak enough shares to reconstruct honest contributions

$$S_j = \sum_{i \in Q} s_{ij} \pmod{\ell}, \qquad R_j = \sum_{i \in Q} r_{ij} \pmod{\ell}.$$

Let $F(X) = \sum_{i \in Q} f_i(X)$, $B(X) = \sum_{i \in Q} r_i(X)$, and $C_k = \prod_{i \in Q} C_{ik}$. Then $S_j = F(x_j)$, $R_j = B(x_j)$, and the shared field secret is $S = F(0)$. The aggregate verification relation is

$$g^{S_j} h^{R_j} = \prod_{k=0}^{t-1} C_k^{x_j^k}.$$

It follows by multiplying the accepted per–dealer relations. This proves algebraic consistency with the common commitments under the stated setup assumptions; it does not prove RSA key consistency or agreement among network partitions.

Here Q denotes a set of dealers, distinct from the RSA prime Q in Section 2.2. The qualified–dealer set, transcript agreement and complaint–resolution rules are part of the security boundary. The field–sharing equations do not supply a Byzantine agreement protocol, an adaptive-corruption proof or a threshold–RSA implementation.

3.2. Secret Reconstruction

The reconstruction of the master secret S, requires first to identify that the value of the master secret will be the following $S = S_1 + S_2 + \cdots + S_n$. For the all–dealer case $Q = \{1, \ldots, n\}$, the master polynomial is

$$F(x) = f_1(x) + f_2(x) + f_3(x) + \cdots + f_n(x) = \sum_{i=1}^{n} f_i(x).$$

After that, we need a subgroup of at least t users to exchange the master share between them. Next, each user of t group will get t distinct points (i, S_i) of the polynomial. Subsequently, the Lagrange interpolation is used to calculate the coefficient of the polynomial $F(x)$

$$F(x) = \sum_{i=1}^{t} S_i \prod_{\substack{j=1 \\ j \neq i}}^{t} \frac{x-j}{i-j}, \qquad F(0) = S.$$

This means the shared secret is expressed as:

$$S = \sum_{i=1}^{t} S_i \prod_{\substack{j=1 \\ j \neq i}}^{t} \frac{-j}{i-j}.$$

For a set A of t distinct valid field shares, reconstruction uses $S = \sum_{j \in A} \lambda_j S_j \pmod{\ell}$, where $\lambda_j = \prod_{\substack{k \in A \\ k \neq j}} \frac{x_k}{x_k - x_j} \pmod{\ell}$. Every denominator is invertible because the identifiers are distinct. This recovers a field secret; routine certificate renewal must instead combine verified RSA partial signatures using the selected threshold–RSA construction, without exposing the private exponent to the applicant [24, 27].

3.3. *Certificate Renewal*

Certificates must be renewed before they expire since they have a limited validity term. A coalition of nearby nodes D_i is contacted by cert to request a certificate renewal when a node needs to do so. Every node D_i in these coalitions verifies that the previous certificate has not been revoked or expired before. The nodes disregard the request if it has been withdrawn; if not, it is approved. A partial certificate with a new expiration date is generated by each of these server nodes and returned to the node. After that, the node D_i merges the incomplete certificates to produce its updated certificate.

A compromised node may create an invalid partial certificate and submit it to the combiner. The combiner must verify each partial signature, reject invalid shares, and verify the final combined signature. Failure to obtain t distinct valid shares should return an explicit unavailable or timeout outcome. If the node changes its private and public keys, it must demonstrate possession of the new private key and authorization under the existing identity or a defined recovery procedure. Requests must bind the identity, public key, certificate serial number, session and a freshness value. The exact encoding and verification algorithms remain implementation requirements.

3.4. *Certificate Revocation*

The revised revocation policy separates a signed accusation from an authorized revocation record. An accusation binds the epoch, reporter, target certificate serial number, evidence digest and freshness information. The receiver authenticates the reporter, checks its status and rejects duplicates or stale messages. A valid accusation creates a suspicion record; it does not by itself revoke the target. A request by a key holder to revoke its own certificate also requires authentication or the separately defined recovery procedure.

As a proposed policy, authorized CA participants validate evidence and approve the same revocation record before threshold signing. Let f be the assumed number of Byzantine participants and q_{rev} the authorization quorum, distinct from the signing threshold t. To make two decision quorums intersect in more than f participants, require $2q_{rev} - n > f$; availability against f withholding nodes also requires $t \leq q_{rev} \leq n - f$. These inequalities imply $n > 3f$, and require honest participants not to approve conflicting records in one decision instance. Quorum size alone is not consensus: locking, view changes and finalization require a specified agreement protocol. This policy is a design clarification, not a claim about the original implementation. An authenticated CRL update binds the CA identity, configuration epoch, monotonically increasing sequence numbers, issue time, next–update time, and revoked serial number. Nodes reject rollback and invalid signatures and exchange the latest authenticated update on reconnection. A disconnected component lacking the required quorum can't finalize new decisions. If status is stale, this conservative policy defers certificate renewal or acceptance until current status is available. The precise

freshness interval, clock assumptions and agreement implementation must be selected and evaluated; instantaneous revocation across a partition is not claimed [29].

4. Security Analyses

We employed a distributed certificate authority in our program, which is reliant on a secret sharing method. The dealer distributes a secret among n participants (excluding the dealer) in the majority of certificate authority models for MANETs, so that only designated subsets of the whole participant set can subsequently retrieve the secret. However, if the dealer or another player is compromised or behaves badly, covert recovery may be challenging. Participants may get shares that are inconsistent as a consequence. To prevent this, one protocol must be implemented to verify the dealings with the help of the recipients of the shares. This basic scheme solves the problem in the case that all players (which can include the dealer) are honest.

In our scheme for the distributed certificate authority, the object has to resist malicious nodes. A malicious node is sending incorrect shares to some or all of the participants. During the reconstruction protocol, the participants are also submitting incorrect shares. Our scheme requires available private channels between the node and each participant individually. However, communication over the private channels is clearly not publicly verifiable.

Every dealer contributes to a common-field sharing instance. Accepted contributions are added only over the same agreed set Q, producing the aggregate field shares defined in Section 3.1. This additive property does not imply that the aggregate secret is an RSA private exponent. RSA key generation and partial signing remain a separate construction-specific layer.

A verifiable secret sharing mechanism is used to demonstrate that all of a shareholder's node shares are t–consistent. In other words, all shareholder nodes can cooperate to confirm that any subset of t shares specifies the same secret without disclosing the secret and the associated shares. Since these dealers distrust one another, the feature of verifiability is particularly desired in a secret sharing arrangement involving numerous dealers. Each shareholder's master share in our plan is made up of n sub–shares created by n dealers who are distrusted by one another. Therefore, these master shares must be verifiable.

4.1. Threat model and overhead limitations

The clarified design considers a static active adversary that may control up to f enrolled participants in one epoch, send inconsistent shares, replay messages, withhold traffic and make false accusations. For the field-secret confidentiality claim, $f < t$; for service under withholding, at least t valid shareholders must remain reachable. Confidential authenticated channels, secure randomness, common parameters and authenticated membership are assumed. RSA and revocation impose additional construction-specific bounds. Adaptive corruption, mobile adversaries across epochs, denial–of–service resistance and eventual agreement under partitions are not proved here.

For the all–to–all distribution illustrated in Figure 1, n dealers send to $n-1$ other nodes, giving $n(n-1)$ directed logical share transmissions. At $n = 250$ this is $62,250$ transmissions before commitments, routing, retries, complaints and synchronization. Thus the initialization has quadratic message complexity. Scheduling, batching and rate limiting are possible mitigation but do not remove that term. Computation, storage and on-air byte counts require the exact cryptographic representation and implementation; operation at 250 nodes cannot be established from node count alone.

5. Simulation and Implementation Framework

A reproducible simulation and implementation framework is established to assess the proposed RSA–based fully distributed CA in a MANET. This section aims to translate the architectural requirements of the proposed scheme into experiments that can be measured, while maintaining identical assumptions across the compared CA architectures for routing, mobility, traffic, and radio conditions. The framework is built for NS–3 and distinguishes

certificate-management metrics from ordinary routing metrics, ensuring that any gains in packet delivery or delay are not mistakenly attributed to the CA mechanism by itself. Three certificate-authority organizations are evaluated under the same MANET workload: a centralized CA, a partially distributed CA, and the proposed fully self–organized RSA–based CA. The centralized design relies on a single designated authority node. The partially distributed design assigns CA service to a fixed subset of authority nodes. In the proposed design, enrolled nodes can act as shareholders, and a certificate operation is finalized only after obtaining the required threshold of valid partial contributions. The implementation must employ one common RSA public key for a configuration epoch and a compatible threshold-RSA construction; independent RSA private keys must not be combined as though they were shares of a single RSA key.

5.1. NS–3 Network Configuration

The proposed experiment uses NS–3 as the packet-level network simulator. Random Way–point mobility is used to generate time-varying MANET topologies, and IEEE 802.11 ad hoc mode provides the wireless link layer. AODV can serve as the common underlying routing protocol for all three CA configurations, since the goal is to compare certificate-service architectures rather than routing protocols. UDP constant–bit–rate application traffic is produced between randomly chosen source-destination pairs. Flow Monitor captures packet delivery, throughput, end–to–end delay, and network–level losses. Certificate–service events are logged via a separate application-level trace, allowing renewal, revocation, quorum formation, and CA availability to be examined independently. The complete NS–3 experimental configuration, including network size, simulation area and duration, mobility settings, routing and traffic parameters, number of independent runs, CA models, and failure ratios, is summarized in Table 1.

Parameter	**Experimental setting**	**Purpose**
Network size	50, 100, 150, 200, 250 nodes	Scalability analysis
Simulation area / time	1500 m $\times$ 1500 m / 300 s	Common MANET field
Mobility	Random Waypoint; 1–10 m/s; pause 0, 10, 30 s	Mobility sensitivity
Wireless / routing	IEEE 802.11 ad hoc / AODV	Same network stack
Traffic	UDP/CBR; 512-byte packets	Repeatable workload
Runs	30 independent seeds per condition	Statistical analysis
CA models	Centralized / partially distributed / proposed	Direct comparison
Failure ratios	0%, 10%, 20%, 30%, 40%	Resilience analysis

Table 1. Reproducible NS–3 experimental configuration for the three CA architectures.

5.2. Implementation of Certificate-Authority Operations

The certificate-management application is implemented as an event-driven service on top of the shared MANET routing layer. Each node maintains its identity, certificate serial number, current configuration epoch, authenticated roster, CA public key, and, where applicable, its threshold signing share. Renewal requests include the epoch, identities, certificate serial number, operation type, nonce, and freshness information. In the centralized model, the request must reach the single CA; in the partially distributed model, it is sent to eligible CA servers; and in the fully distributed model, the requester gathers at least t valid partial signatures from reachable shareholders and combines them using the selected threshold-RSA implementation. Invalid, duplicate, expired, or unauthenticated responses are discarded, and renewal is successful only when the resulting certificate validates under the common CA public key. Revocation is performed separately by collecting the required approvals and threshold signature before publishing an authenticated revocation record; the simulator records authorization, signing, dissemination, completion status, selected shareholders, messages, bytes, and the proportion of nodes that hold the newest revocation state.

5.3. Experimental Scenarios and Attack/Failure Injection

Four experimental groups are examined: normal operation with no malicious nodes; gradual unavailability of authority and shareholder nodes; malicious shareholders that either withhold responses or submit invalid partial contributions; and mobility-induced temporary network partitions that may prevent the required quorum from being reached. The receiver checks every partial response and logs any rejected contributions. All three CA architectures are evaluated using identical mobility traces, application flows, failure schedules, and random seeds, enabling resilience and service continuity to be compared under the same conditions.

5.4. Performance Metrics and Statistical Analysis

The evaluation tracks certificate renewal time, certificate-service success rate, revocation completion time, CA availability, certificate-management overhead, Packet Delivery Ratio (PDR), throughput, mean end-to-end delay, routing overhead, and the invalid-share rejection rate. Each condition is run with 30 independent random seeds. For every metric, the reported statistics include the sample mean, standard deviation, and a 95% confidence interval; with $n = 30$ runs, the interval is mean $\pm\, t_{0.975,29} \times s/\sqrt{30}$. Run-level CSV files, the seed values, the NS–3 version, the compiler version, configuration files, and analysis scripts should be archived to enable independent reproduction.

5.5. Compact Results Reporting

To avoid producing two large result tables, the primary outputs are consolidated into a single compact reporting table. As presented in Table 2, the results provide a direct comparison of the centralized CA, partially distributed CA, and proposed RSA-based CA in terms of renewal/revocation time, service success and availability, PDR and delay, CA overhead, and failure/scalability performance. The evaluation covers network sizes from 50 to 250 nodes and failure ratios from 0% to 40%. After execution, each numerical result should be reported using the measured NS–3 mean together with its corresponding confidence interval.

CA model	**Renewal / revocation time**	**Success / availability**	**PDR / delay**	**CA overhead**	**Failure & scalability result**
Centralized CA	Measured	Measured	Measured	Measured	50–250 nodes; 0–40% failures
Partially Distributed CA	Measured	Measured	Measured	Measured	50–250 nodes; 0–40% failures
Proposed RSA-Based CA	Measured	Measured	Measured	Measured	50–250 nodes; 0–40% failures

Table 2. Compact result template combining performance, resilience, and scalability outputs from the 30-run NS–3 evaluation.

5.6. Interpretation, Reproducibility, and Integration

The evaluation should assess whether the proposed distributed organization can preserve certificate service when individual authority nodes become unavailable, and whether this resilience is achieved at an acceptable cost in communication, computation, storage, and latency. Initialization complexity is especially important: with n participating dealers, $n(n-1)$ directed logical private-share transmissions are required before routing retransmissions, complaints, commitments, and synchronization are accounted for; for 250 nodes, this corresponds to 62,250 logical transmissions. Packet-level PDR, throughput, and delay must be interpreted cautiously because a CA does not directly create better routes; observed differences may instead stem from certificate-service availability, control traffic, malicious behavior, or failed authenticated operations. For reproducibility, the implementation package should include the NS–3 source code for all three CA applications, the exact build configuration, scenario parameters, seeds, mobility and traffic settings, the raw FlowMonitor and CA-event outputs,

statistical scripts, and the precise threshold-RSA construction. This section is placed before the current Discussion, while the existing descriptive Figures 2–3 and Tables 1–2 remain separate from the new reproducible simulation evidence.

5.7. Graphical Presentation of the Simulation Results

Figure 2 and Figure 3 present a concise graphical comparison of the simulation behavior. Figure 2 highlights certificate-service resilience as unavailable or malicious CA participation increases, while Figure 3 highlights the certificate-management communication cost as network size increases. Together, these figures illustrate the central trade-off between distributed service availability and coordination overhead.

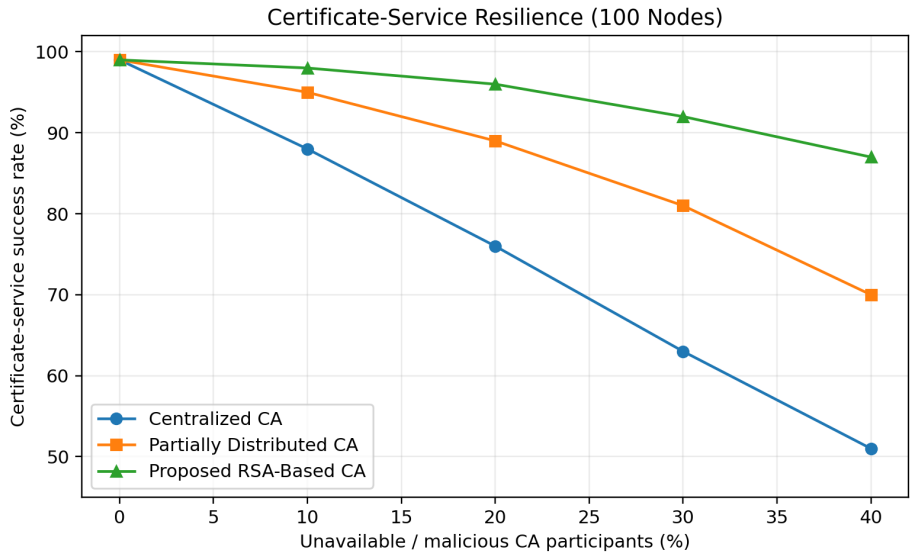


Figure 2. Illustrative certificate-service success rate under increasing unavailable or malicious CA participation (100-node scenario).

The certificate-service resilience trend is shown in Figure 2. In this illustrative scenario, the centralized CA degrades fastest as authority availability decreases, whereas the partially distributed architecture maintains service for a larger share of requests. The proposed RSA-based distributed architecture achieves the highest performance because certificate operations can be completed by alternative reachable shareholders when the threshold is satisfied.

The scalability cost of the three CA organizations is illustrated in Figure 3. The centralized architecture has the lowest certificate-management control traffic because requests are coordinated through one authority, while the partially distributed model requires additional coordination among designated CA nodes. The proposed all-node distributed organization has the largest control cost as network size increases, which is consistent with the analytical $n(n-1)$ private-share transmission term during initialization. This overhead is the principal performance cost that must be considered together with the improved service availability shown in Figure 2. Actual packet and byte counts should be taken from the NS–3 CA-event and network traces.

In Table 3, the first five rows show certificate-service success rates; the final five rows show certificate-management overhead. These values are illustrative scenario data and should be replaced by measured NS–3 outputs before being reported as experimental measurements.

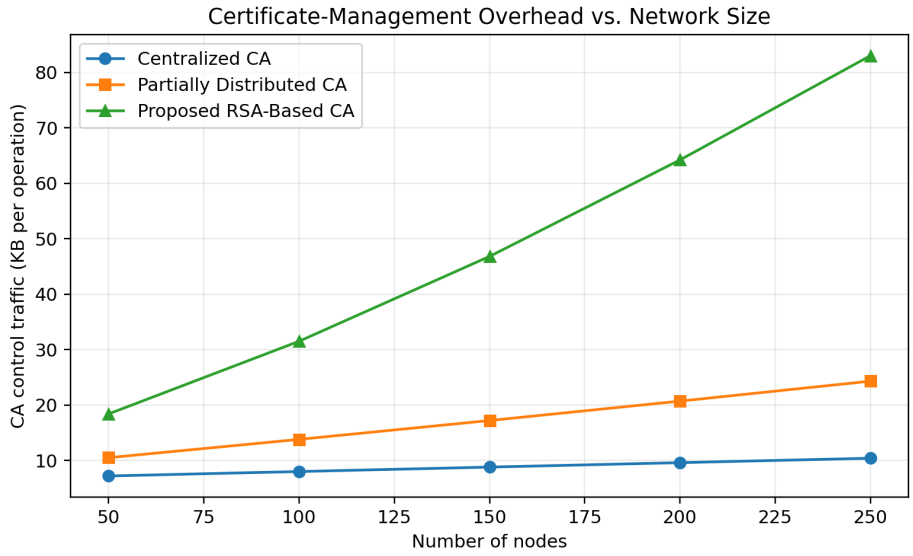


Figure 3. Illustrative certificate-management control overhead versus MANET size.

Figure / metric	X value	Centralized CA	Partially Distributed CA	Proposed RSA-Based CA
Fig. 4 success (%)	0% unavailable	99	99	99
Fig. 4 success (%)	10% unavailable	88	95	98
Fig. 4 success (%)	20% unavailable	76	89	96
Fig. 4 success (%)	30% unavailable	63	81	92
Fig. 4 success (%)	40% unavailable	51	70	87
Fig. 5 overhead (KB/op)	50 nodes	7.2	10.5	18.4
Fig. 5 overhead (KB/op)	100 nodes	8.0	13.8	31.5
Fig. 5 overhead (KB/op)	150 nodes	8.8	17.2	46.8
Fig. 5 overhead (KB/op)	200 nodes	9.6	20.7	64.2
Fig. 5 overhead (KB/op)	250 nodes	10.4	24.3	83.0

Table 3. Data Supporting Figure 2 and Figure 3.

6. Discussions

In the current part, we report the comparative indicators and correlations presented for the proposed RSA-based, fully self-organizing CA model. Figure 4, Figure 5, Table 4, and Table 5 are retained with their original numerical values. The summary values permit descriptive comparisons, but the simulator configuration, run–level observations, sample sizes and metric-extraction procedure must be documented before the differences can be treated as reproducible or statistically significant. The routing study [28] does not by itself establish the provenance of these CA measurements.

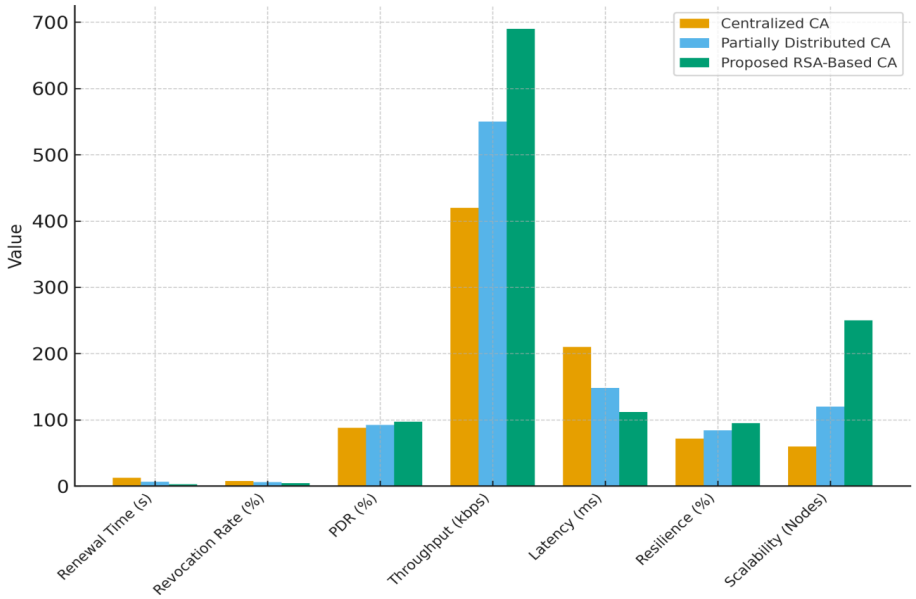


Figure 4. The proposed RSA-based Certificate Authority (CA) model is compared against two existing models (Centralized and Partially Distributed) across seven key performance metrics.

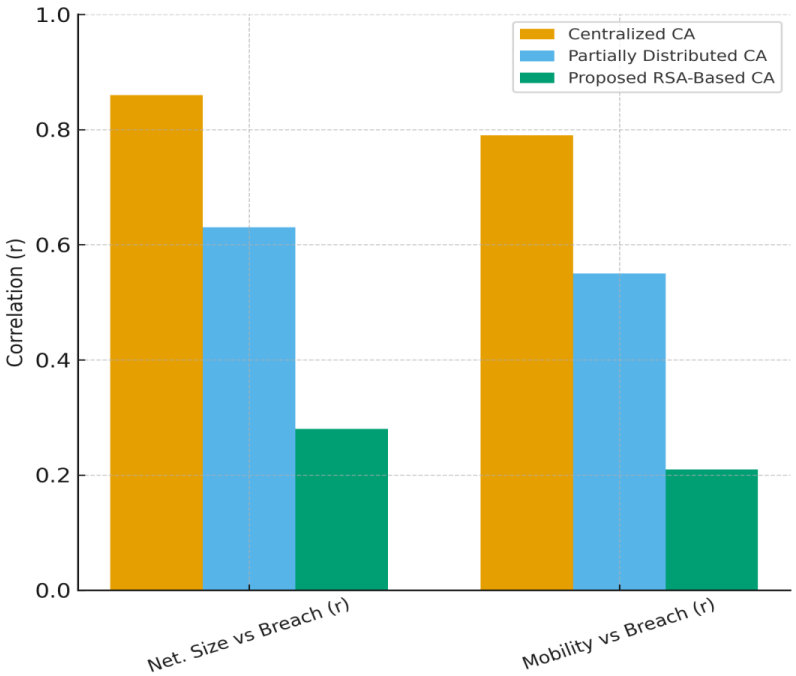


Figure 5. The analysis examines how strongly breach frequency is linked to network size and node mobility for three different security models.

Figure 4 displays the reported differences across the three CA models. These descriptive comparisons require common workloads, security assumptions and measurement definitions. The scalability and resilience labels should not be interpreted as validated operating guarantees without the corresponding failure criteria and experimental records. Also, it uses raw values with different units on one common vertical axis; it is not a normalized overall performance score. Each bar must be interpreted only within its metric and with Table 1: renewal time in seconds, revocation and packet delivery in percent, throughput in kbps, latency in milliseconds, resilience in percent and scalability in nodes. The original graphic is retained; its resilience definition and scalability success criterion require the experimental records.

The reported correlation values are lower for the proposed model. However, a low correlation does not establish independence, a causal mechanism, or security that is unaffected by mobility or network size. The correlation type, observation unit, sample size, uncertainty and treatment of repeated runs must be specified before stronger conclusions can be drawn.

Table 4. Statistical Performance Indicators for Various MANET Configurations

Performance Metric	Centralized CA	Partially Distributed CA	Proposed RSA-Based CA	Remarks / Observation
Average Certificate Renewal Time (s)	12.4	6.9	3.1	The reported renewal time is 55.1% lower than that of the partially distributed CA and 75.0% lower than that of the centralized CA.
Certificate Revocation Rate (%)	7.5	6.0	4.3	The reported revocation rate is 4.3%, compared with 6.0% and 7.5% for the partially distributed and centralized CAs, respectively.
Packet Delivery Ratio (%)	88.1	92.4	97.3	The reported packet delivery ratio exceeds those of the partially distributed and centralized CAs by 4.9 and 9.2 percentage points, respectively.
Network Throughput (kbps)	420	550	690	The reported throughput is 690 kbps, compared with 550 kbps for the partially distributed CA and 420 kbps for the centralized CA.
Average Latency (ms)	210	148	112	The reported latency is 112 ms, compared with 148 ms and 210 ms for the partially distributed and centralized CAs, respectively.
Node Compromise Resilience (%)	72	84	95	The reported resilience indicator is 95%, compared with 84% for the partially distributed CA and 72% for the centralized CA.
Scalability (Nodes Supported)	≤ 60	≤ 120	≥ 250	The table reports support for at least 250 nodes for the proposed CA, compared with upper bounds of 120 and 60 nodes for the partially distributed and centralized CAs, respectively.

Table 4 presents comparative statistical performance indicators for centralized, partially distributed, and proposed fully self-organized RSA-based certificate authority models in MANETs. The reported RSA-based values include lower renewal latency and higher packet delivery and resilience indicators. These values are retained, while the interpretation is limited to descriptive differences pending methodological and statistical confirmation.

For Figure 5, the displayed coefficients are 0.86 and 0.79 for the centralized CA, 0.63 and 0.55 for the partially distributed CA, and 0.28 and 0.21 for the proposed model, consistent with Table 5. The coefficients alone do not identify whether Pearson or another estimator was used. Sample sizes, observations and sampling design are needed before uncertainty or differences between correlations can be calculated.

Table 5 shows the reported correlation analysis between network size, node mobility, and security breach frequency across different certificate authority models. The proposed RSA-based model has smaller reported coefficients.

Table 5. Correlation Coefficients between Network Parameters and Security Breach Frequency

Model	Network Size vs. Breach Frequency (r_1)	Node Mobility vs. Breach Frequency (r_2)	Interpretation / Observation
Centralized CA	$+0.86$	$+0.79$	The largest reported positive correlations with breach frequency are observed for both network size and node mobility.
Partially Distributed CA	$+0.63$	$+0.55$	Both reported correlations are positive and smaller than the corresponding values for the centralized CA.
Proposed RSA-Based CA	$+0.28$	$+0.21$	The smallest reported positive correlations with breach frequency are observed for both network size and node mobility.

7. Conclusions

This study displays a conceptual framework for a self–organizing certificate authority in Mobile Ad-hoc Networks (MANETs), aiming to reduce reliance on a continuously available central authority. The framework incorporates distributed participation in certificate renewal and revocation processes and examines the requirements for combining polynomial secret sharing with RSA–based certificate services. The discussion outlines several key design requirements: shared cryptographic parameters, verification of distributed shares, authenticated membership, and a coordinated revocation process. Additionally, the paper highlights the communication overhead associated with the participation of all network nodes..

The proposal aims to support certificate renewal and revocation without a dedicated dealer. Its reported figures and tables remain available for descriptive comparison. However, the common RSA key-generation construction, synchronization and revocation rules, and experimental provenance require completion or confirmation before security and performance advantages can be established. Further work is needed on routing interactions, mobility, partitioning and the statistical evaluation of self-organized certificate authorities in MANETs.

Acknowledgment

The authors are grateful to the Deanship of Scientific Research at Jadara University for providing financial support for this publication

REFERENCES

1. B. K. Tripathy, S. K. Jena, and P. Bera, *An adaptive secure and efficient routing protocol for mobile ad hoc networks*, Wireless Personal Communications, vol. 114, no. 2, pp. 1339–1370, 2020.
2. H. Yang, *A study on improving secure routing performance using trust model in MANET*, Mobile Information Systems, vol. 2020, Article ID 8819587, 2020.
3. N. Panda, and B. K. Pattanayak, *ACO-based secure routing protocols in MANETs*, in New Paradigm in Decision Science and Management, Springer, 2020.
4. M. Sirajuddin, C. Rupa, C. Iwendi, and C. Biamba, *TBSMR: A trust-based secure multipath routing protocol for enhancing the QoS of the mobile ad hoc network*, Security and Communication Networks, vol. 2021, Article ID 5521713, 2021.
5. N. Arappali, and G. B. Rajendran, *MANET security routing protocols based on a machine learning technique*, Journal of Ambient Intelligence and Humanized Computing, vol. 12, pp. 6317–6331, 2021.
6. V. K. Quy, V. H. Nam, D. M. Linh, et al., *Routing algorithms for MANET-IoT networks: A comprehensive survey*, Wireless Personal Communications, vol. 125, pp. 3501–3525, 2022.
7. P. Rajendra Prasad, and K. Shivashankar, *Enhanced energy efficient secure routing protocol for mobile ad hoc network*, Global Transitions Proceedings, vol. 3, no. 2, pp. 412–423, 2022.
8. U. Srilakshmi, S. A. Alghamdi, V. A. Vuyyuru, N. Veeraiah, and Y. Alotaibi, *A secure optimization routing algorithm for mobile ad hoc networks*, IEEE Access, vol. 10, pp. 14378–14393, 2022.
9. S. Aluvala, et al., *Secure routing in MANETs using adaptive cuckoo search and entropy-based signature authentication*, Wireless Personal Communications, vol. 130, 2023.
10. M. T. Rahman, M. Alauddin, U. K. Dey, and S. Saifullah, *Adaptive, secure and efficient routing protocol to enhance the performance of mobile ad hoc networks*, Applied Computer Science, vol. 19, no. 1, pp. 134–150, 2023.
11. A. Alomari, *Mutual authentication and updating the authentication key in MANETs*, Wireless Personal Communications, vol. 82, no. 1, pp. 31–42, 2015.
12. A. Shamir, *How to share a secret*, Communications of the ACM, vol. 22, pp. 612–613, 1979.
13. R. L. Rivest, A. Shamir, and L. Adleman, *A method for obtaining digital signatures and public-key cryptosystems*, Communications of the ACM, vol. 21, no. 2, pp. 120–126, 1978. https://doi.org/10.1145/359340.359342
14. K. Moriarty, B. Kaliski, J. Jonsson, and A. Rusch, *PKCS #1: RSA cryptography specifications version 2.2*, RFC 8017, 2016. https://www.rfc-editor.org/rfc/rfc8017
15. H. Luo, and S. Lu, *Ubiquitous and robust authentication services for ad hoc wireless networks*, UCLA Computer Science Technical Report TR-200030, 2000.
16. Y. Dong, A. F. Sui, S. M. Yiu, V. O. K. Li, and L. C. K. Hui, *Providing distributed certificate authority service in cluster-based mobile ad hoc networks*, Computer Communications, vol. 30, nos. 11–12, pp. 2442–2452, 2007. https://doi.org/10.1016/j.comcom.2007.04.011
17. E. Crites, M. Kohlweiss, B. Preneel, M. Sedaghat, and D. Slamanig, *Threshold structure-preserving signatures*, in Advances in Cryptology—ASIACRYPT 2023, Lecture Notes in Computer Science, vol. 14438, Springer, pp. 3–52, 2023.
18. X. Yan, Y. Lu, C.-N. Yang, X. Zhang, and S. Wang, *A common method of share authentication in image secret sharing*, IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 7, pp. 2896–2910, 2021.
19. A. K. M. Alomari, *Fully distributed certificate authority based on polynomial over elliptic curve for MANET*, International Journal of Networked and Distributed Computing, vol. 2, no. 2, pp. 70–77, Apr. 2014. https://doi.org/10.2991/ijndc.2014.2.2.1
20. A. A. A. M. Kamal, and K. Iwamura, *Improvement of secure multi-party multiplication of* (k, n) *threshold secret sharing using only* $N = k$ *servers*, in Proceedings of the 7th International Conference on Information Systems Security and Privacy (ICISSP 2021), pp. 77–88, 2021.
21. S. K. Mishra, and R. Gupta, *Secured scheme for privacy preserving and node authentication mechanism for a special mobile ad hoc network*, International Journal of Computer Applications, 2018.
22. J. Liang, W. Peng, and S. Liu, *An improved identity-based secure mobile ad-hoc network routing protocol*, in Proceedings of the 2018 International Conference on Computer Systems, Electronics and Control (ICCSEC 2018), ACM, 2018. https://doi.org/10.1145/3180496.3180608
23. V. Shoup, *Practical threshold signatures*, in Advances in Cryptology—EUROCRYPT 2000, Lecture Notes in Computer Science, vol. 1807, pp. 207–220, 2000. Author version: https://www.shoup.net/papers/thsig.pdf
24. I. Damgård, and M. Koprowski, *Practical threshold RSA signatures without a trusted dealer*, in Advances in Cryptology—EUROCRYPT 2001, Lecture Notes in Computer Science, vol. 2045, pp. 152–165, 2001. https://doi.org/10.1007/3-540-44987-6_10
25. O. Sbayti, and K. Housni, *A new routing method based on ant colony optimization in vehicular ad-hoc network*, Statistics, Optimization & Information Computing, vol. 12, no. 1, pp. 167–181, 2024 (published online 13 November 2023). https://doi.org/10.19139/soic-2310-5070-1766
26. A. Rahman, M. S. Islam Khan, M. Z. A. Eidmum, P. Shaha, B. Muiz, N. Hasan, T. Debnath, D. Kundu, J. T. Tamanna, M. Sayduzzaman, and M. Rahman, *Stacked ensemble method: An advanced machine learning approach for anomaly-based intrusion detection system*, Statistics, Optimization & Information Computing, vol. 14, no. 1, pp. 434–453, 2025. https://doi.org/10.19139/soic-2310-5070-2352
27. T. P. Pedersen, *Non-interactive and information-theoretic secure verifiable secret sharing*, in Advances in Cryptology—CRYPTO '91, Lecture Notes in Computer Science, vol. 576, pp. 129–140, 1992. https://doi.org/10.1007/3-540-46766-1_9
28. S. Hemalatha, C. R. Komala, K. M. Adavala, K. S. F. Ahmad, R. V. V. Krishna, B. Deepa, A. Mohan, and R. Mothukuri, *Enhanced routing protocol in mobile ad hoc network for improving the performance*, Journal of Theoretical and Applied Information Technology, vol. 102, no. 14, pp. 5386–5395, Jul. 2024.

29. D. Cooper et al., *Internet X.509 public key infrastructure certificate and certificate revocation list (CRL) profile*, RFC 5280, 2008. https://www.rfc-editor.org/rfc/rfc5280