

Distributed Certification Authority for Mobile Ad Hoc Networks – A Dynamic Approach

Sanjay Raghani, Durga Toshniwal, R. C. Joshi
Department of Electronics & Computer Engineering
Indian Institute of Technology - Roorkee, Roorkee, India
sanjay.raghani@gmail.com, {durgafec, joshifcc}@iitr.ernet.in

Abstract

A Mobile Ad Hoc Network (MANET) is an infrastructureless network of wireless mobile nodes that cooperate among each other to maintain connectivity of the network. In comparison to wired networks, securing a MANET is more difficult and challenging. One of the effective ways of providing security in MANETs is by using Public Key Cryptography and Certificates. Certificates are issued by a trusted entity known as Certification Authority (CA). In order to provide certification service in MANETs, researchers have proposed the design of a distributed CA based on threshold cryptography. With a distributed CA, nodes obtain certification service from a coalition of threshold number of nodes in the network. In the design of distributed CA that we follow, nodes identify a coalition within their one-hop neighbors. With a distributed CA, when the number of nodes in the network reduces, number of neighbors for nodes also reduces resulting in substantial increase in the certification service delays. In this paper we propose the design of a distributed CA which follows dynamic approach to solve the above problem using a suite of network monitoring protocols. The proposed design is based on the observation that when the number of nodes in the network reduces, the delays involved in obtaining certificates can be reduced by dynamically varying the threshold value.

1. Introduction

In recent years, there has been an increase in interest for research in ad hoc networks because of the large requirement of infrastructureless networking in various application domains. Due to the absence of infrastructure in MANETs, routing is performed by the participating nodes using in the network. Few of the applications of MANETs include military networks, disaster area networks, collaborative and distributed computing, wireless sensor network, wireless mesh networks and hybrid wireless networks [1]. Importance of MANET has also been identified by Internet Engineering Task Force (IETF) which has formed a working group “manet” to promote research in this area of networks [2]. In comparison to wired networks, securing a MANET is more difficult because of the absence of infrastructure, use of shared broadcast channel, limited battery and computing power, node mobility and dynamic joining and leaving of nodes [1, 3, 4]. Moreover, the security requirements of MANETs, like other networks, include authentication, confidentiality, integrity, availability and non repudiation [1]. These requirements can be satisfied by using public key cryptography and certificates [5, 6]. Certificates are issued by a Certification Authority (CA) which is a trusted by all the members of the network.

In infrastructure based networks, centralized design of CA is preferred. Centralized CA, however, cannot be used in MANETs, as it would require fixed infrastructure support and the scalability and availability of the network will also be reduced. Due to the constrained network environment of MANETs, researchers have proposed solutions for distributing the functionality of CA amongst the nodes in the network [3, 4, 7, 8]. These solutions use threshold cryptography [9, 10] to distribute the ability to provide cryptographic services amongst the nodes in the network. threshold cryptography is based on secret sharing schemes [11]. With a distributed CA, certification service is now provided by a coalition of threshold number of nodes. The threshold value is a critical parameter and is decided based on the security requirement and number of nodes in the network. In the design of distributed CA that we follow, the functionality of a CA is distributed amongst all the nodes in the network and nodes obtain certification service by communicating with their one-hop neighbors [4, 7]. These schemes rely on the assumption that the number of neighbors for each node, also known as node degree, remains higher than the threshold value throughout the network lifetime. In actual scenarios, the number of nodes in the MANET varies with time and the above assumption does not hold true throughout the network lifetime. We also observe that with this approach, when the number of nodes in the network reduces, there is a significant increase in

the certification service delay. Further it degrades the overall performance of distributed CA as: (1) Nodes require more time for renewing the certificate. (2) Number of requests issued by a node increases, resulting in an increase in the network traffic. (3) Few nodes may be unable to renew their certificates within the validity period and thereby will be ceased from further communication in the network.

In this paper we propose the design of a distributed CA which follows dynamic approach and addresses the above problem using a suite of network monitoring protocols. The proposed protocol suite extends the design of a distributed CA by providing dynamic behavior. The protocol suite enables a distributed CA to dynamically adjust the threshold value, when required, and thereby results in reduction of certification service delays. We provide simulation results to support the effectiveness of the proposed scheme. The rest of the paper is organized as follows. Section 2 briefly describes functionality of a distributed CA along with an overview of the existing work. Section 3 presents system design for the proposed scheme. Section 4 covers the architecture of the proposed distributed CA with dynamic approach, and describes the working of each sub protocol of the proposed protocol suite. Section 5 covers threshold cryptographic details in reference to the proposed protocols. Functionality of each protocol is described using mathematical equations. Section 6 covers the simulation details along with the performance evaluation of the proposed scheme. Section 7 is the conclusion of the paper.

2. Background

Public key cryptography and certificates can be used for providing security in wired as well as wireless networks. Kerberos [12] and standard X.509 [13] are popular network authentication architectures for wired networks that are based on certificates. Many solutions for securing MANETs have also been proposed that use public key cryptography and certificates [3, 4, 5, 7, 8]. A certificate for a node includes the identity of a node, public key of the node, validity period and other details of the node and is signed using the private key of the CA. Nodes contact the CA to obtain a new certificate, renew a certificate or to obtain the public key of other nodes in the network. Certificates can be verified by any other nodes in the network using the public key of the CA. A typical example of a certificate for a node (v_i) will be:

CA $\rightarrow v_i$: cert $_{v_i} = [v_i, K_{v_i+}, t_1, t_2] K_{CA-}$. (1) where K_{CA-} is the private key of CA, K_{v_i+} is the public key of node v_i , t_1 is the time of issuing the certificate and t_2 is the validity period of the certificate. Once a node obtains a certificate, it can use it for a fixed period of time, equal to the validity period of the certificate. The validity period depends on the security requirement of the system. Each node must periodically renew its certificate or obtain a new certificate before an existing certificate expires. A node must send the certificate renewal request to the CA, at proper time so as to renew the certificate before an existing certificate expires. In MANETs, distributed approach for CA is preferred in comparison to the centralized approach. Distributed CA is based on the concept of threshold cryptography which distributes the ability to provide cryptographic service among the nodes by distributing a share value corresponding to the cryptographic key [9]. Distributed CA distributes the signing functionality of CA amongst nodes in the network. The share values distributed among the nodes correspond to the private key of CA. Centralized approach of CA is also a special case of distributed CA with threshold = 1.

One of the earliest attempts of using threshold cryptography in MANETs was carried out by Zhou and Haas [3]. Zhou and Haas in [3] distribute the functionality of the CA among a subset of nodes in the network. Kong, et al. in [4] distributes the functionality of CA among all the nodes in the network improving scalability and availability of the service. Yi, et al. in [8] deals with heterogeneous networks and recommends distribution of the functionality of CA among selected nodes that are computationally more powerful and secure. Royer et al. in [5], uses public key cryptography and certificates at network layer and has proposed a secure routing protocol for MANETs. A threshold cryptographic system can tolerate presence of maximum *threshold - 1* number of corrupted players, provided, sufficient number of uncorrupted players exists to continue the service. An adversary needs to compromise at least threshold number of players in order to retrieve the private key of CA and breach the security of the system. To further improve the security of the networks using distributed CA, proactive schemes [14] have been proposed. Proactive schemes allow periodic refreshing of share values of the nodes. As the share values are refreshed, the time window available to an adversary to compromise the security of the system is reduced. An adversary now needs to compromise threshold number of nodes within one time period.

In most of the designs of distributed CA that uses threshold cryptography, the threshold value remains constant throughout the network lifetime [3, 4, 7]. In practical scenarios, a change in the threshold value is required when: (1) The security requirements of a network changes (2) Number of nodes in the network changes. In the first scenario, one needs to define a policy to identify the security requirement of a network and cause a corresponding change in the threshold value. In the second scenario, 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. In this paper, we study the second scenario and propose

the design of distributed CA with dynamic approach, which uses a suite of network monitoring protocols to avoid the above problem.

The problem of change in the threshold value has been studied earlier but not much work has been done in the context of MANETs. Blakley et al. in [15] provides solutions for change in threshold value using Broadcast channels. Zhou and Haas in [3] and Desmedt et al. in [16] discusses solutions for change in threshold value in the presence of secure channels. Most of the existing work only describes the ways in which threshold value can be modified. Not much work has been done to identify the situations during network operations, when a change in the threshold value should actually happen. In the proposed design of CA, we follow the approach of Desmedt et al. [16] and carry out an appropriate change in threshold, when required so as to prevent an increase in certificate renewal delay. The proposed scheme monitors the Average Node Degree of the network to identify the situations for change in threshold value.

3. System Design

Our proposed design of distributed CA with dynamic approach, uses a suite of network protocols to monitor the average node degree of the network. The protocol suite comprises of three protocols and a monitoring scheme viz. (1) Certificate renewal protocol. (2) Neighbor discovery protocol. (3) Node degree monitoring scheme and (4) Protocol for change in threshold value. As mentioned earlier, for basic functionality of CA, we follow the design of distributed CA as proposed in [4, 7] wherein a node obtains the certification service from its one hop neighbors. We evaluate the performance of the protocols by studying the effect of reducing the number of nodes in a MANET on the certificate renewal delay. Our scheme relies on the following assumptions:

- One of the nodes in the network, the leader node, always remains up throughout the network life time and calculates Average Node Degree of the network
- Presently we do not consider any adversary model for the network
- Changes in the number of nodes in a network are slow
- Proposed protocol suite adjusts the threshold value of the network. We assume that the security requirement of the system can be satisfied with the new threshold.
- Nodes follow Random Direction Mobility Model [18, 19].

First assumption requires existence of a leader node which helps in initiating the change in threshold value. Leader node itself possesses no additional secret information and system security will remain intact even if an adversary is able to compromise the leader node. In the absence of the leader node, system continues to operate but with a degraded performance. In practical scenarios the leader node may not remain alive throughout the network lifetime. In order to handle this, leader node can be monitored and a substitute can be elected as a leader when an existing leader node goes down. We used RSA [17] algorithm for public key cryptography but the proposed scheme is also valid for any public key cryptography scheme. The absence of the infrastructure and limited battery power refrain use of complex communication protocols in MANETs due to which we used connectionless, unreliable protocol at transport layer. Wherever reliability was required, ATCP [20] like approach was used and reliability was provided at higher layers. For routing, we used Ad Hoc On Demand Distance Vector Routing protocol (AODV) [21] a popular reactive routing protocol for MANETs. Presently we follow a simple approach for removing the nodes from the network and make the nodes inactive in a linear pattern over a period of time.

4. Architecture

In this section, we describe the architecture of the proposed distributed CA with dynamic approach. Working of CA is explained by covering the functionality of each sub protocols of the proposed protocol suite. With a distributed CA, each node possesses a share of the private key of CA corresponding to the threshold value. These nodes are called initialized nodes. During network setup phase, a minimum of threshold number of nodes are initialized by providing the share of private key of CA. These nodes can further initialize other nodes of the network.

4.1 Certificate Renewal Protocol

In certification based security schemes, each node needs to renew its certificate before the existing certificate expires. With a distributed CA, a node can renew its certificate by communicating with threshold number of initialized nodes in its neighborhood. Initialized nodes are the nodes which possess share of private key of CA. Any coalition of threshold

number of initialized nodes can provide certification service to other nodes in the network. A coalition of threshold number of initialized nodes can retrieve the private key of CA using their share values and is thus able to provide certification service to other nodes. However during the certificate renewal process, none of the nodes becomes aware of the private key of CA as the key is not reconstructed at any of the participating nodes.

In order to renew a certificate, the requesting node first identifies a coalition of threshold number of initialized nodes in its neighborhood. Requesting node then broadcasts certificate details along with the coalition information to the members of the coalition. If the requesting node is an initialized node, it can also be a part of the coalition. Coalition members on receiving the certificate details, calculates the partial certificates using their share values and reply to the requesting node. Requesting node on obtaining partial certificates from all the coalition members combines them to generate an actual certificate. Actual certificate is encrypted using CA's private key. As mentioned earlier, the private key of CA is not revealed to any participant of the network during the certificate renewal process.

If the degree of a node is small, relative to the threshold value, a node will be unable to identify a coalition of threshold number of initialized nodes in its neighborhood. In such cases the requesting node will rebroadcast the certificate renewal request after a timeout period. As the nodes are mobile, neighborhood of a node changes continuously. After the timeout, a node might be able to identify a coalition of threshold number of initialized nodes with the new set of neighbors and subsequently renew its certificate. A node may also timeout multiple times, before a successful certificate renewal. It has been observed that the delay experienced in certificate renewal is much higher if Average Node Degree of the network is less than the threshold value.

4.2 Neighbor Discovery Protocol

Each node runs the Neighbor Discovery Protocol using which it periodically calculates its one hop neighbors, also known as degree of the node. In order to obtain the node degree, a node broadcasts HELLO packet with TTL (time to live) field set to 1. Due to the wireless environment and mobility of nodes, node degree values calculated using Neighbor Discovery protocol may not be accurate and the actual number of neighbors of a node may be higher than the calculated value. However the node degree value calculated using the Neighbor Discovery protocol reflects the number of replies a node can successfully receive from its neighbors and so it can be used as an estimate for the threshold value of the network.

The period for Neighbor Discovery protocol, depends on the requirements of the system. Smaller periods would result in accurate node degree value but it would also increase the control traffic packets in the network. On obtaining the response from the neighbors, each node calculates the degree value and communicates it to the leader node. Leader Node uses the node degree values of individual nodes to calculate the Average Node Degree of network.

4.3 Node Degree Monitoring

In the proposed scheme, one of the nodes in the network, Leader Node periodically calculates the Average Node Degree of the network. It uses the node degree values obtained from other nodes in the network to calculate average node degree of the network. Due to collisions and packet loss in the network, leader node may not receive node degree values from all the nodes in the network. In such cases, leader node calculates the average degree over the number of values received. As nodes follow Random Direction Mobility Model [18, 19], the variation in node degree value for each node is small over the network lifetime. Due to this the Average Node Degree calculated over the response received can be used as the Average Node Degree for the entire network. When the number of nodes in the network reduces, leader node uses the Average Node Degree value of the network to decide the new threshold. When the Average Node Degree of the network becomes less than the threshold value, certificate renewal requests, would timeout and nodes have to issue multiple requests before a successful certificate renewal. In order to prevent this, leader node initiates a change in the threshold value of the network when the Average Node Degree falls below the current threshold value. In our design, leader node initiates a change in the threshold value after observing lower Average Node Degree for two consecutive time periods. The new value of threshold is calculated as follows:

If (Avg. Node Degree < T_{curr}) for two consecutive periods
 $T_{new} = \max (T_{min}, \tau * \text{Avg. Node Degree})$

where τ is the multiplicative factor ($0 < \tau < 1$)
 T_{curr} = Current threshold value
 T_{min} = Minimum value of threshold (equal to 2)

T_{new} = New threshold value

Value of multiplicative factor depends on the security requirements of the system and must be less than or equal to one. Use of multiplicative factor less than one prevents frequent changes in threshold value when the number of nodes in the network reduces continuously. We selected a typical value of 0.9 for our experiments.

4.4 Change in Threshold

When the leader node observes a lower value of Average Node Degree of the network for two consecutive time periods, it initiates a Change in the Threshold value of the network. Change in Threshold value occurs in two stages a) Localized Change in Threshold b) Network Wide Change in Threshold. The second stage, network wide change is initiated only after ensuring successful completion of the first stage.

4.4.1 Localized Change in Threshold

This is the first stage of change in threshold value. In this stage, leader node first initiates change in its neighborhood and causes at least T_{new} number of nodes to change their share values of private key of CA, to a new value corresponding to the new threshold (T_{new}). To initiate this, leader node first forms a coalition of T_{curr} number of initialized nodes which agree to change the threshold value. Leader node can also be a part of the coalition. Nodes use secure communication among the coalition members for change in threshold value. We refer to the participating nodes during Localized Change in Threshold as special nodes. Assuming the leader node also participates in the coalition, it first identifies a coalition of T_{curr} special nodes. On identifying a coalition, leader node transmits the new threshold value and coalition details to the all the members of the coalition. Each special node then exchanges new partial shares of their existing share values, corresponding to the new threshold (T_{new}). Special nodes upon receiving the partial shares from all the other special nodes, calculates the new share value of CA's private key and notifies the leader. Leader node on receiving notification from at least T_{new} number of nodes notifies them to switch to the new share values. Special nodes acknowledges the leader node on changing their share values and again the leader node requires at least T_{new} number of acknowledgements before it starts the next stage of Network Wide Change in Threshold. This two step process is required to ensure that sufficient number of nodes successfully update their share values of private key of CA so that they can later initialize other nodes in the network.

4.4.2 Network Wide Change in Threshold

During Network Wide Change in Threshold, leader node sends a broadcast message to all the nodes in the network to update their share values of private key of CA. All the nodes update their share values by communicating with their neighbors which have already updated their shares. The second stage is similar to the Distributed Self Initialization Protocol as discussed in [7]. Thus the proposed protocol suite enables a distributed CA to update the threshold value by monitoring the average node degree of the network and thereby prevents an increase in certificate renewal delay.

5. Threshold Cryptography

Threshold cryptography distributes the ability to provide cryptographic service among the members of a group [9]. A (t , n) threshold cryptography scheme allows any t out of n nodes to jointly perform a cryptographic operation whereas any coalition of less than t nodes is unable to perform the operation [9]. In this section, we explain the cryptography operations with reference to the functionality of a distributed CA.

5.1 Network Initialization

Network Initialization is performed during the start of network. In this stage, nodes obtain the share value of the private key of CA. Nodes which obtain the share are called initialized nodes. Network Initialization is performed with the help of a special node termed as dealer node [4, 7]. During network initialization, dealer node first generates RSA keys [17] for CA. Let the keys generated by the dealer node be:

Public key: (e , n)

Private key: (d , n)

where e is RSA Public exponent
 d is RSA Private exponent
 n is the RSA modulus.

Dealer node then defines a $k-1$ degree polynomial for sharing the private key of CA as:

$$f(X) = \sum_{i=0}^{k-1} a_i X^i \quad (2)$$

where $a_0 = d$. and a_1, a_2, \dots, a_{k-1} are distributed over a finite field.

The value k represents the threshold value of the network. Dealer node then initializes at least k nodes by providing them with partial shares of CA's private key d . After initializing sufficient number of nodes in the network, dealer node goes offline due to security reasons [4]. A node with identity v_i obtains its share of private key of CA, P_{v_i} , from dealer node as:

$$P_{v_i} = f(v_i) \bmod n \quad (3)$$

A node can also obtain its share from a coalition of k initialized nodes $\{v_1, v_2, \dots, v_k\}$ using Lagrange's Interpolation formula as:

$$P_{v_i} = f(v_i) \bmod n = \left(\sum_{j=1}^k l_{v_j}(v_i) P_{v_j} \right) \bmod n \quad (4)$$

$$\text{where } l_{v_i}(x) = \prod_{j=1, j \neq i}^k \frac{(x - v_j)}{(v_i - v_j)}$$

A coalition of k initialized nodes $\{v_1, v_2, \dots, v_k\}$ can also recover CA's private key as:

$$d = f(0) = \left(\sum_{i=1}^k l_{v_i}(0) P_{v_i} \right) \bmod n \quad (5)$$

5.2 Threshold RSA Signatures

Any coalition of threshold number of initialized nodes can generate a valid signature of CA's private key. When a node needs to renew its certificate, it first identifies a coalition of initialized nodes in its neighborhood as discussed in Section 4.1. Requesting node then broadcasts to the coalition members, the certificate ($cert$) which is to be signed along with the coalition information. Certificate of the requesting node $cert$ should be signed using the private key CA (d) as:

$$CERT = cert^d \bmod n \quad (6)$$

Coalition members on receiving the $cert$ value, generate partial certificate by signing it with their additive shares. Additive share SK_{v_j} of a node v_j is calculated as:

$$SK_{v_j} = P_{v_j} l_{v_j}(0) \bmod n \quad (7)$$

Node v_j then calculates the partial certificate for the requesting node using the additive share value as:

$$CERT_{v_j} = cert^{SK_{v_j}} \bmod n \quad (8)$$

Node v_j then transmits the partial certificate to the requesting node. Requesting node on receiving all the partial certificates combines them by multiplication to generate a candidate certificate $CERT'$ as:

$$CERT' = \prod_{j=1}^k CERT_{v_j} \quad (9)$$

$CERT'$ may differ from actual certificate $CERT$ by an additional exponent which is k -bounded multiple of n . $CERT$ can be recovered from $CERT'$ by applying the “ k -bounded coalition offsetting algorithm” proposed by Kong et al. [4].

5.3 Change in Threshold

As mentioned earlier, Change in Threshold occurs in two stages. The leader node first identifies a coalition of threshold number of initialized nodes. Coalition members first carry out Localized Change in Threshold and then participate in Network Wide Change in Threshold. Without loss of generality assume v_i is the leader node and v_2, v_3, \dots, v_k are the nodes which along with v_i form a coalition to change the threshold value of the network from an existing value of k to k' . On forming a coalition, leader node first communicates the new threshold value along with the coalition information to all the members of the coalition. Each coalition member v_j then defines a secret sharing scheme [11] in which its share of CA's private key (P_{v_i}) is shared using a (n, k') threshold scheme. Each node v_i first defines a $k'-1$ degree polynomial as:

$$f_{v_i}(X) = \sum_{j=0}^{k'-1} a_j X^j \quad (10)$$

where $a_0 = P_{v_i}$ and $a_1, a_2, \dots, a_{k'-1}$ are distributed over a finite field.

After forming such polynomial, each node v_i securely exchanges the share value corresponding to the secret P_{v_i} with other members of the coalition. A coalition member node v_j receives its share from node v_i as:

$$P_{v_j, v_i} = f_{v_i}(v_j) \bmod n \quad (11)$$

When the node v_j receives shares from all other coalition members, it updates its share of CA's private key to a new value corresponding to the new threshold k' as:

$$P_{v_j}' = \left(\sum_{i=1}^k l_{v_i}(0) P_{v_j, v_i} \right) \bmod n \quad (12)$$

$$\text{where } l_{v_i}(0) = \prod_{r=1, r \neq i}^k \frac{v_r}{v_r - v_i} \bmod n$$

A minimum of k' number of nodes in the coalition must be able to update their share value from P_{v_i} to P_{v_i}' before proceeding to the Network Wide Change in Threshold. Once the required number of nodes successfully updates their share value of private key of CA corresponding to the new threshold, the leader node issues a notification to the entire network for Network Wide Change in Threshold. A node v_i can then obtain its share of private key of CA corresponding to the new threshold (k') by forming a coalition of threshold (k') number of newly initialized nodes as discussed in Section 5.1.

6. Simulation and Results

We evaluated the performance of the proposed design of CA by simulating the protocols in NS2 [22], a discrete event network simulator in Linux environment. The cryptographic routines were written in C++ using OpenSSL [23], an open source cryptographic library. The results were obtained using a Pentium 4, 2.4 GHz System with 256 MB RAM. In order to evaluate the efficiency of the proposed protocols, we used the following performance metrics:

- Average Certificate Renewal Delay – Measures the average latency for each node to carry out a successful certificate renewal.
- Average Number of Attempts - Average value of the number of attempts a node requires before a successful certificate renewal.

- Time Delay for Change in Threshold - Total time required for the network to update its threshold value to a new threshold value after the leader node initiates a change in threshold.

Important parameters used for the simulation of the proposed protocols in NS2 are shown in Table 1.

Table1. Simulation Parameters

Parameter	Value
Number of Nodes	100
Total Simulation Time	500sec , 800 sec
Area of Network	1000mts * 1000 mts
Routing Protocol	AODV
Threshold Value	10
% of Inactive nodes	20, 30, 40, 50
Velocity of nodes	10 m/s
Mobility Model	Random Direction
Key Size	512 bits – 2048 bits

Table 2. Threshold RSA Delay Values

Key Size (msec)	Encryption (msec)	Decryption (msec)	Partial Cert Comp. (msec)	Comb. Partial Cert. (msec)
512	0.054	1.681	5.619	0.402
768	0.076	4.256	9.589	0.600
1024	0.113	13.524	19.762	0.640
1280	0.135	29.668	22.147	1.191
1536	0.179	49.992	34.441	1.838
1792	0.213	80.316	50.788	1.823
2048	0.305	83.303	77.256	2.273

6.1 Threshold RSA

The results of Threshold RSA are shown in Table 2 varying the key size. We used these delay values in NS2 for simulation of the proposed protocols. These delay values may differ from the optimum benchmark, but the overall pattern of the results would remain same even if the optimum values were used. In Table 2, Encryption and Decryption refers to the standard RSA Encryption and Decryption delays. Partial Cert Comp. value is the delay experienced by a coalition member (initialized node) in calculating the partial certificate for a requesting node. Comb. Partial Cert. is the delay experienced by the requesting node in combining threshold number of partial certificates. From Table 2 we observe that the computational delay for RSA increases with an increase in the key size. However these delay values are small in comparison to the communication delay experienced by the nodes.

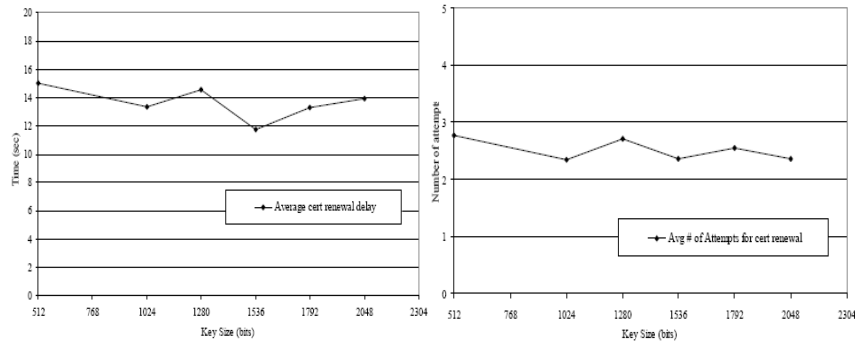


Figure 1. Avg. Cert. renewal delay v/s Key Size Figure 2. Avg. # of attempts v/s Key Size

6.2 Effect of Key Size on Certificate Renewal

Figure 1. and Figure 2. show the effect of varying the key size on the certificate renewal service. From the figure we observe that when the key size increases, the average number of attempts and Average Certificate Renewal Delay remains almost constant. The reason for such behavior is that the communication delays in a wireless network are much higher in comparison to the cryptographic delays. Therefore the average certification delay value is dominated by the communication delay of the network. Also as the computations for partial certificate are carried out in parallel, delay for combining the partial certificates is negligible. Therefore an increase in key size doesn't result in an increase in certificate renewal delay or number of attempts.

6.3 Neighbor Discovery Protocol

Figure 3 shows the results obtained using the Neighbor Discovery Protocol. Timely variations in Average Node Degree value as calculated by the leader node are shown in the figure. Results were obtained by varying the number of nodes in the network. For all the cases, nodes were made inactive following a linear pattern, between time 250s and 300s. We observe that when the number of inactive nodes is zero, the Average Node Degree of the network remains almost constant. This is because nodes follow Random Direction Mobility Model [18, 19]. In all the other cases, when the number of nodes in the network reduces, node degree value of each node reduces resulting in reduction of the Average Node Degree value of the network.

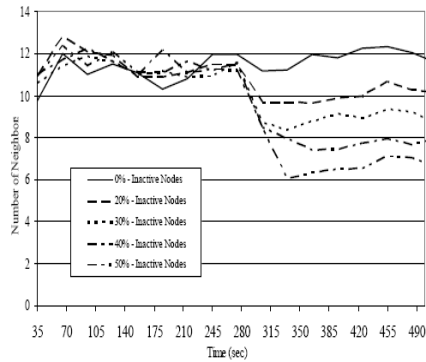


Figure 3. Avg. Node Degree of Network v/s Time

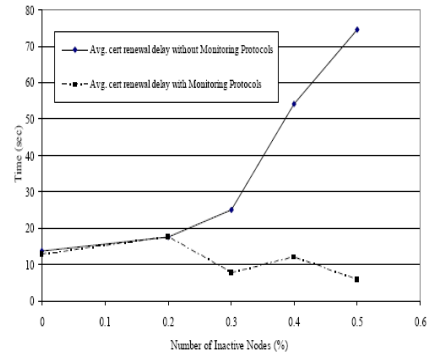


Figure 4. Avg. Certificate Renewal Delay v/s

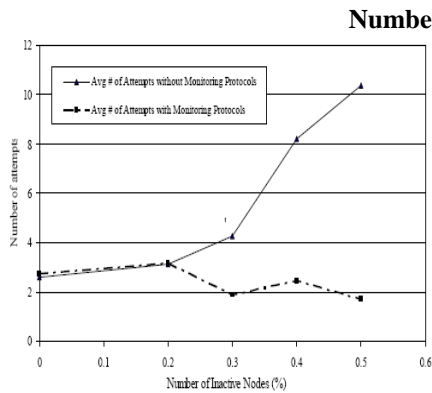


Figure 5. Avg. # of attempts v/s Number of Inactive Nodes

Number of Inactive Nodes

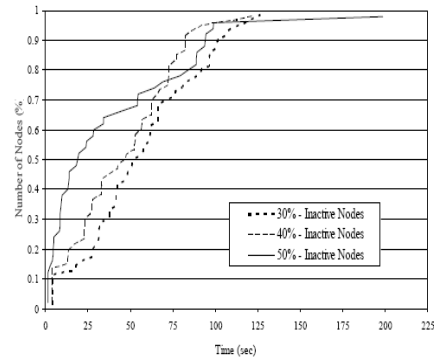


Figure 6. Number of nodes (%) v/s Time Delay for Change in Threshold

6.4 Effect of Variation in Number of Nodes on Certificate Renewal

Figure 4 and Figure 5 show the effect of reducing number of nodes in the network on the certificate renewal service. Initial threshold value was taken as 10 and RSA Key length was taken as 1024 bits. Average Certificate Renewal Delay and Average Number of Attempts required for certificate renewal has been shown by inactivating 20%, 30%, 40% and 50% of nodes in the network.

From Figure 4 and Figure 5, we observe that when the number of inactive nodes increases, there is a substantial increase in the Average Certificate Renewal Delay. Due to an increase in certificate renewal delay, nodes may be unable to renew the certificate within the validity period. Also it becomes difficult for a node to decide the time of sending certificate renewal request. Average number of attempts in renewing a certificate also increases substantially, when the number of inactive nodes increases. We also observe that when the Average Node Degree of a network is less than or equal to threshold, most of the nodes are able to renew the certificate in their first attempt. Few nodes still experienced large delays and attempt (timeout) several times before a successful renewal thereby resulting in Average Number of Attempts greater than 1.

When the proposed suite of monitoring protocols is used, the values for Average Certificate Renewal Delay and Average Number of Attempts are less as the monitoring protocol prevents an increase in these values by adjusting the threshold. When the number of inactive nodes is large, the achieved reduction in Average Certificate Renewal Delay and Average Number of Attempts using the proposed scheme is significant.

6.5 Time Delay for Change in Threshold

Figure 6 shows the time it takes for the entire network to update the threshold value. It is the total time during which all the nodes in the network obtain new share value of private key of CA, corresponding to the new threshold. For simulation, the number of nodes in the network was taken as 100, initial threshold value was fixed to 10, and RSA key length was taken as 1024 bits. Time Delay for Change in Threshold has been shown by inactivating 30%, 40% and 50% of nodes in the network.

6.5.1 Localized Change in Threshold

In Figure 6, we observe that the curve of “Time Delay for Change in Threshold” increases vertically in the beginning. This corresponds to the Localized Change in Threshold during which nodes in the neighborhood of the leader node update their share values. As the leader node uses broadcast message to notify the special nodes to update share values, all special nodes update their shares at almost same time. This results in an increase in the number of initialized nodes at the same time, resulting in the observed pattern of graph.

6.5.2 Network Wide Change in Threshold

The rest of the curve of “Time Delay for Change in Threshold” corresponds to the network wide change in the threshold value. During this period, all the nodes in the network obtain new share values of private key of CA, corresponding to the new threshold. This stage is similar to the distributed self initialization protocol [7], so the total time required for the entire network to change the threshold value is less. In practical scenarios, the validity period of a certificate is few minutes, so the Time Delay for Change in Threshold of a network is acceptable as the threshold gets updated in less than the validity period of a certificate.

7. Conclusion & Future Work

In this paper we discussed the design of a distributed CA for MANETs based on threshold cryptography. We found that the delay experienced by nodes for certificate renewal increases when the number of nodes in the network is reduced. We proposed a set of monitoring protocols which extends the design of a distributed CA by providing dynamic behavior. The protocols enable the distributed CA to dynamically update the threshold value by monitoring the Average Node Degree of the network and thereby prevent an increase in the certificate renewal delay. Using the proposed design, we have achieved a significant reduction in certificate renewal delay and in the number of attempts required for certificate renewal. Simulation results verify the effectiveness of our proposed scheme.

The proposed design of distributed CA with dynamic approach can be further extended for supporting networks with frequent changes in number of nodes. Extending the design for considering adversary model for the network will also be an interesting area for future research.

References

- [1] C. Siva Ram Murthy, B.S. Manoj, "Ad Hoc Wireless Networks : Architectures and Protocols", Prentice Hall PTR, May 2004, New Jersey.
- [2] Internet Engineering Task Force Mobile Ad hoc Network (MANET) Working Group, <http://www.ietf.org/html.charters/manet-charter.html>.
- [3] L. Zhou and Z. J. Haas. Securing ad hoc networks. IEEE Network Magazine, 13(6), 1999.
- [4] J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang. Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks. In Proceedings of ICNP '01.
- [5] K. Sanzgiri, B. Dahill, B. Levine, C. Shields, E. Belding-Royer, "A Secure Routing Protocol for Ad Hoc Networks", In Proceedings of 2002 IEEE International Conference on Network Protocols (ICNP), November 2002.
- [6] W. Stallings "Cryptography and Network Security", Prentice Hall, India, 2004.
- [7] H. Luo, S. Lu, Ubiquitous and Robust Authentication Services for Ad Hoc Wireless Networks, UCLACSDTR-200030.
- [8] S. Yi and R. Kravets. MOCA: Mobile certificate authority for wireless ad hoc networks. In 2nd Annual PKI Research Workshop (PKI03), April 2003.
- [9] Y. Desmedt and Y. Frankel. Threshold cryptosystems. In CRYPTO, 1989.
- [10] V. Shoup. Practical threshold signatures. In EUROCRYPT, 2000.
- [11] A Shamir, "How to share a secret," Communications of ACM, 1979.
- [12] J. Kohl and B. Neuman, "The Kerberos network authentication service (version 5)," RFC-1510.
- [13] Arsenault and S. Turner, "Internet X.509 public key infrastructure," draft-ietf-pkixroadmap-06.txt, 2000.
- [14] Herzberg, S. Jarecki, H. Krawczyk, and M. Yung, "Proactive secret sharing, or: How to Cope with Perpetual Leakage", in Proc. of CRYPTO 95. Extended abstract, 1995.
- [15] B. Blakley, G.R. Blakley, A. Chan and J. Massey. Threshold schemes with Disenrollment. Adv. In Cryptology - CRYPTO'92, Lecture Notes in Comput. Sci., 740, (1993), 540-548.
- [16] Desmedt, Y. and Jajodia, S. Redistributing secret shares to new access structures and its applications. Technical Report ISSE TR-97-01, George Mason University, July 1997.
- [17] R. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems", Communications of the ACM, Feb. 1978
- [18] Elizabeth M Royer, P. M. Melliar-Smith, Louise E. Moser, "An Analysis of the Optimum Node Density for Ad hoc Mobile Networks", Proceedings of the IEEE International Conference on Communications, Helsinki, Finland, June 2001.
- [19] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research," Wireless Communication and Mobile Computing, vol2, no. 5, pp. 483-502, 2002.
- [20] J. Liu and S. Singh, "ACTP: Application Controlled Transport Protocol for Mobile Ad Hoc Networks," Proceedings of IEEE WCMC 1999, vol. 3, pp. 1318-1322, September 1999.
- [21] C. E. Perkins and E. M. Royer, "Ad hoc On-Demand Distance Vector Routing," in Proc. WMCSA, Feb. 1999
- [22] NS-2 (The Network Simulator). <http://www.isi.edu/nsnam/ns/>.
- [23] OpenSSL Project, <http://www.openssl.org/>.

Sanjay Raghani is a graduate student in the Department of Electronics & Computer Engineering at Indian Institute of Technology - Roorkee, Roorkee, India. He completed his B.E in Information Technology from Nirma Institute of Technology, Ahmedabad, India. His areas of interests include ad hoc networks, cryptography, network security and computer architecture.

Durga Toshniwal is Assistant Professor in the Department of Electronics & Computer Engineering at Indian Institute of Technology - Roorkee, Roorkee, India. She received her Ph.D degree from Indian Institute of Technology, Roorkee, India in 2000. Her areas of interests include ad hoc networks, databases and data mining.

R. C. Joshi is Professor in the Department of Electronics & Computer Engineering at Indian Institute of Technology - Roorkee, Roorkee, India. He received his B.E. degree in Electrical Engineering from Allahabad University in 1967. He then received his M.E. and Ph.D. degrees in Electronics and Computer Engineering from Indian Institute of Technology - Roorkee (formerly University of Roorkee) in 1970 and 1980, respectively. His areas of interests include computer networks, parallel & distributed processing, artificial intelligence and database.