Adaptation of a Conference Key Distribution System for the Wireless Ad Hoc Network

C. Tselikis, S. Mitropoulos, C. Douligeris
Department of Informatics
University of Piraeus
80 Karaoli & Dimitriou Str., Piraeus 185 34, Greece
ctselik@gmail.com

G. Tselikis
Ministry of Education, Athens, Greece
Papandreou 37, Marousi, 15180, Greece
gtselikis@minedu.gov.gr

N. Komninos
Department of Computer Science
City, University of London, EC1V 0HB
nikos.komninos.1@city.ac.uk

***Abstract*— In this paper we review previous works done with respect to Conference Key Distribution Systems (CKDS). We focus on the system proposed by Kim et al. and we propose improvements on that scheme *a)* from the perspective of security and anonymity, *b)* from the perspective of efficient calculation of the Lagrange polynomial coefficients, and *c)* from the perspective of adaptation into the dynamic wireless ad hoc network. The security of the proposed scheme is based on the difficulty of computing discrete logarithms over elliptic curves, the intractability of inverting a one-way hash function and the pseudo-randomness of user coordinates. We demonstrate the effectiveness of the proposed scheme through the analysis of characteristic attack scenarios**.

Keywords—Shared secret; Lagrange interpolation; elliptic curves;

I. INTRODUCTION

Security and anonymity play vital role for group communications in environments such as the Internet, Mobile Ad hoc Networks (MANET), Wireless Sensor Networks (WSN) and the new generation networks, like Multi-hop Cellular Networks (MCN) and 5G networks (Cloud-RAN). To exemplify, in the infrastructure-less wireless ad hoc network many group applications, such as content sharing, are protected with a symmetric key. Also, in Multi-hop Cellular Networks the Base Station and the terminals which are located in different cells have to cooperate for the establishment of a symmetric encryption key while keeping anonymity.

In this paper, we focus on a specific category of key distribution protocols, namely Conference Key Distribution Systems (CKDS) with chairperson. Such CKDS appeared quite long before to secure group sessions with the calculation of a Conference Key (CK) by a chairperson who distributes the security parameters to the session participants. We aim to anonymize further the CKDS and exploit it in a more dynamic environment, namely the multi-clustered wireless ad hoc network. Our aim is to establish a CK in order to secure the communications inside each cluster which consists of a number simple cluster members and one Cluster Head (CH). However in this case we have to take into account that in the highly dynamic ad hoc conditions very frequently a new cluster head (chairperson) appears and also the cluster members may change under the same CH due to mobility, and hence the security parameters have to change accordingly.

This work was partially funded by the University of Piraeus.

A. Motivation

Our motivation is to review a past series of works done on three-stage CKDS and to identify the weaknesses for each examined scheme. The elegant work done by Kim et al. [6] was a strong incentive for us to optimize the ECDLP-based Lagrange polynomial interpolation method utilized in CKDS. In addition, we are extending our work of [17] to anonymise and optimize further the calculation and distribution of a CK in the MANET environment.

B. Contributions

- We offer a concise review of the works and the comments we encountered in the literature regarding CKDS with chairperson, especially those prior and including the Kim et al. scheme. We also cite our own comments..
- We improve the Kim et al. scheme from the anonymity and security vulnerabilities perspective. Our proposed scheme does not require any node identification information during the key generation phase as it makes use of a polynomial equation system with Langrage interpolation and pseudorandom node coordinates. Thus we avoid linking attendant identities with their private keys.
- We offer details regarding the calculation of the Lagrange polynomial coefficients.
- We adapt the three-stage CKDS in a way that it is suitable for the dynamic wireless ad hoc environment. The security of our scheme is based on the intractability of the Elliptic Curve Discrete Logarithm Problem (ECDLP), the hardness of inverting hash functions and the use of ephemeral (rather than long-term) public keys during the ECDH key agreement

phase. Moreover, we employ numerical analysis methods in order to calculate the necessary security parameters.

This paper is organized as follows. In section II we examine previous works and in section III we cite our comments for the Kim et al. scheme. In section IV we cite the proposed scheme and in Section V we present a performance analysis summary for the two schemes. Section VI examines re-keying issues in CKDS and in Section VII we analyze the security of our scheme. Finally, section VIII concludes and states the future work.

II. Related Work

The CKDS concept with chairperson was introduced in [1]. In [2] T.C Wu proposed a CKDS with user anonymity based on algebraic approach with the use of one-way hash functions to hide the identities of the attendants. In [3] the three-stage Tseng-Jan scheme proposed user anonymity with two improvements on Wu's CKDS, however being based on the same cryptographic assumptions. The first improvement proposed a CKDS with simple polynomial interpolation and the use of hash function to hide the participants identities and random selection of the conference key (CK) by the chairperson. The second proposal did not use a one-way function. In [4] the Yang, Chang and Hwang scheme proposed user anonymity based on the intractability of the ECDLP. In their three-stage scheme the conference key is randomly chosen by the chairperson who then broadcasts to the attendants the values y_i that belong to a linear curve.

In [5] the Lin et al. commented the Yang et al. scheme [4] and described an "intruder" attack in which a non-legitimate user (i.e., someone who does not share a secret with the chairperson) could, according to the authors claims, uniquely solve a set of linear equations to acquire the h_i of the legitimate participants and also to recover CK. To overcome that attack of solving equations simultaneously, Lin et al. proposed a small-cost modification of the Yang et al. scheme in which the transmitted values y_i are substituted by $y_i' = h_i \oplus y_i$.

Kim et al. [6] reviewed the Yang et al. scheme [4] and the Lin et al. scheme [5]. They identified a weakness common in both schemes, namely that a legitimate attendant to distinguish his own value y_i, without receiving any useful information from the chairperson, should check all y_i values to recover CK and verify its validity which incurs unnecessary computational cost. To overcome, and in better support of user anonymity, Kim et al. proposed a three-stage CKDS scheme which for the made use of the Lagrange polynomial interpolation method. In the Kim et al. scheme [6] the chair-person calculates the CK which has to be recovered by the legitimate attendants by means of the Lagrange coefficients which are broadcasted in a broadcast message M.

In [7] Tang and Mitchell, almost concurrently with Kim et al., commented the Yang et al. [4] and the Lin et al. [5] schemes. They correctly stated that Lin et al., in their intruder attack definition against the Yang et al. scheme, failed to observe that the system of equations the "intruder" attacker has to solve to obtain the hashes h_i and CK is actually *n x (n+1)* and not *n x n* and therefore their solution is not unique. In addition, in their comments on the Yang et al. scheme Tang and Mitchell identified a different vulnerability. Namely, they identified one attack in which a legitimate attendant, i.e., someone who shares a secret with the chairperson, after recovering CK, can solve *n x n* system of equations to obtain all h_i values of the rest legitimate attendants. In sequence, the attacker pretending the chairperson can forge a new message M sending modified values $y_j' = c_1(h_j) + CK'$ thus including a new, valid but forged conference key. To our view this is a correct vulnerability identification regarding the Yang et al. scheme and in the rest of this paper we will call this type of attack the *intruder* attack. The second weakness identified by Q. Tang and C. J. Mitchell [7] is the same with that spotted by Kim et al, namely that there is no real anonymity supported in [4] and [5] since a legitimate attendant has to determine his own value y_i, or y_i', of all values received in order to solve for the CK.

A summary of the comments we found in the literature regarding Yang et al., Lin et al. ([4] and [5]) schemes follows:

- Both schemes do not really maintain the user anonymity since the values y_i or y_i' distributed to the attendants are directly linked to their identity.
- Trying to keep anonymity in Yang et al. and Lin et al. schemes would lead to unnecessary computation costs for key recovery and key verification by all the attending users, as underlined by Kim et al in [6].
- Both schemes are vulnerable to the *intruder* attack in which an attendant attacker can obtain h_i values of the rest legitimate participants and hence can forge new CK to mislead them.

Our additional comment regarding schemes [4] and [5] is the following:

Yang et al. and Lin et al. assume that the private keys x_i are randomly assigned by the *system* to the nodes and delivered to them through a *secure channel*, which is unsafe because increases the chances to solve the ECDLP. On the contrary, we propose private keys generated *on demand* by an assymetric cryptosystem. The private keys are kept inside each of the attendant nodes. The limitation in this case is the computational cost to generate those keys.

Finally, our work in [17] incorporated the polynomial interpolation approach into ad hoc cluster head selection procedure implemented with voting among the members.

III. A SECOND SERIES OF WORKS AND COMMENTS THAT WE FOUND IN THE LITERATURE RELATES TO A DIFFERENT CATEGORY OF PROTOCOLS, NAMELY FOUR-STAGE KEY AGREEMENT PROTOCOLS WHICH INCLUDE ONE ADDITIONAL FAULT-DETECTION STAGE FOR INTRUDER IDENTIFICATION AND EXCLUSION. INDICATIVELY WE CITE HERE SOME RELATIVE WORKS DONE IN [8], [9], [10], [11], [12] AND [13]. IN [10] WE ENCOUNTERED THE ALREADY IDENTIFIED WEAKNESS OF HAVING TO LINK THE IDENTITY OF PARTICIPANTS WITH THE BROADCASTED SHARES D_i. THIS CATEGORY OF KEY AGREEMENT PROTOCOLS IS OUT OF THE CURRENT PAPER'S SCOPE AND THEREFORE WILL BE STUDIED THOROUGHLY IN OUR FUTURE WORK. COMMENTS ON THE KIM ET AL. SCHEME

1) Initiative stage

Kim et al. assume a system that coincides with a static chairperson who assigns to each session attendant his random private key x_i and chairperson's public key Q_c through a secret channel. On the contrary, we propose that the private-public key pairs must be generated *on demand* by the attendants during the key distribution stage (not during the initiative stage) as also *on demand* must be generated the session keys k_{ic} shared between an attendant and the chairperson. Another solution would be to pre-load the public keys of all users during the initiative stage. However it would dramatically increase the storage requirements for each user. Besides, pre-loaded public keys are long-term keys lacking the ephemeral characteristic of short-term keys that we seek in the dynamic infrastructure-less wireless ad hoc network.

2) Key distribution stage

a) In the Kim et al. scheme the chairperson U_c generates the coordinate pair $\{h_i, H(h_i)\}$ for each participant. The pair-wise session key k_{ic} is protected inside the hash value $h_i = H(k_{ci} \,||\, ID_c \,||\, ID_i \,||\, T) \,||\, m$. Therefore, in the calculation of hash values h_i (Step2 and Step3 in their key distribution stage) Kim et al. expose the participant identity IDi which contradicts to privacy. We claim that the user identities can be used a priori if an authenticated ECDH protocol is adopted to establish the pair-wise session keys, so that there is no need to re-use them when calculating the hash values. In our scheme the hash value is calculated by omitting the participant identity, $h_i = H(k_{ci} \,||\, Z_c \,||\, T) \,||\, m$ so that no opportunity is given to conspiracy and identity attackers to link identities with keys (see section VII).

b) Kim et al. do not provide any forward secrecy guarantees. On the contrary, we propose that the pair-wise session keys k_{ic} are established by forcing each participant and U_c to generate and exchange short-time public keys according to the ECDH protocol during the key distribution stage. Afterwards, the short-time EC-paired keys have to be deleted. In addition, we recommend adopting authenticated DH protocol in the establishment of session keys to guarantee forward-secrecy in CKDS.

c) In Step 3 the chairperson applies Lagrange polynomial interpolation to construct a polynomial of degree n-1 and calculates its n coefficients $(c_{n-1}, \ldots, c_0)$ where CK= c_0 assuming that n is the number of the participants. However, Kim et al. do not specify the way in which the Lagrange coefficients are calculated. We provide details for efficient calculation of the Lagrange polynomial coefficients utilizing *Newton's symmetrical functions*.

d) Kim et al. scheme uses the polynomial interpolation approach. In the case that n is small (i.e., in realistic scenarios for example five attending nodes) then the Lagrange polynomial would have a very small $n-1$ $degree$ and could be analyzed by attacker. In addition, with a poorly designed Elliptic Curve Cryptosystem (ECC) it could be easily solved by an attacker. Clearly, this is another trade-off: the larger the polynomial degree, the more the necessary calculations that have to be performed by legitimate users, however the scheme becomes more secure.

e) In the Kim et al. scheme neither an intruder nor a non-attending attacker can obtain the pairs $\{h_i, H(h_i)\}$ of the legitimate attendants simply because h_i are not distributed by the chairperson. Moreover, for intruder who has recovered CK, to find another attendant's private key it is hard because has to solve the ECDLP. However, if this is achieved, then intruder will have found k_{ic} and then by brute force attack against one-way function could find the corresponding attendant identity. In this way he could totally break the anonymity of the system (knowledge of who of the attendants owns a specific private key and, moreover, who owns a specific session key).

3) Key recovery stage

Same procedure with that described in the Kim et al. scheme.

IV. PROPOSED CKDS SCHEME

A. System assumptions

We change the assumptions made in the Kim et al. scheme; we make necessary adaptations for the wireless ad hoc network. In more detail:

- The security of the communications medium is that of an authenticated broadcast channel. We assume digital signatures so that an unauthorized external attacker cannot learn the key and decrypt the exchanged messages during a session.
- The system can be mainly attacked by malicious internal users (*intruders*) who launch active attacks such as *conspiracy*, *forge of CK pretending the chairperson (impersonation)*, and *anonymity breaking* attack (see section VII).
- In our adaptation of CKDS for the homogeneous wireless ad hoc environment we assume that the network is structured in dynamic clusters, each consisting of member nodes and a Cluster Head who calculates the Conference Key and broadcasts the necessary parameters for the cluster members to recover the key. . In the case that the CH changes, the CKDS procedure has to be restarted.

B. Implementation details

We follow the nomenclature of a (t, n)-threshold secret-sharing scheme (TSS = (PG, DS, SC)) which consists of three

stages: the public parameter generation (PPG), the dealer setup (DS) and the share combiner (SC) to distribute a shared secret CK. Let $A = \{U_1, U_2, \dots, U_m\}_i$ denote the set of all m nodes in the network and let $B = \{U_1, U_2, \dots, U_n\}_i$ denote the set of all legitimate participants in the session ($n < m$).

The PPG stage takes as input a security parameter pair $k \in K$ (here K denotes the secret set $\{0,1\}^K$) and returns a string $y \in Y$ of public parameters (here Y denotes the public set $\{0,1\}^Y$).

The DS stage takes as input a security/public parameter pair (k, y) and a secret s from the secret space $S(k, x) \subseteq \{0,1\}^{k+1}$ and returns a list of n shares $s = (s_1, \dots, s_n)$, where s_i is the ith share space $S(k, x)$ for $i = 1, \dots, n$.

The SC stage takes as input a security/public parameter pair (k, y) and any subset $s_I = \{s_i : i \in I\}$ of t out of the n shares, and returns a recovered secret $s \in S(k, x)$ (here I denotes a subset of [n] of size $\#I = t$). The correctness and security parameters properties of a $(t, n$ -threshold secret-sharing scheme can be quantified by the definitions of those in [14].

Public Parameter Generation (PPG)

We assume that the *EC* public domain parameters (elliptic curve E defined over a finite field F_q with base point G of prime order p) are already known to the ad hoc network. Initially, a chairperson (CH), to start the session, broadcasts initiation message to make known his own public key Q_c to the participants $U_i \in B$.

Dealer Setup (DS)

Each attendant after receiving the EC domain parameters and Q_c secretly generates pseudo-random integer coordinates, $x_i, y_i \in [1, p-1]$ that define a point Z_i. Then each attendant generates hisephemeral public key $Q_i = x_i G$ on the Elliptic Curve Cryptosystem (ECC) and then signs and unicasts this ephemeral public key to the initiating node U_C.

After receiving and authenticating all Q_i, U_c computes the ECDH pair-wise keys $k_{ci} = Z_c Q_i$ shared with each U_i. This is different than [6] in which all Q_i are created and kept inside *the system* (U_c). Then the chairman computes the hash values $h_i = H(k_{ci} \,||\, Z_c \,||\, T) \,||\, m$ and constructs a polynomial with degree $n-1$ using n coordinate points $\{h_i, H(h_i)\}$ by applying Langrage polynomial interpolation, similar to [6] and [14].

$$f(Z) = \sum_{k=1}^{n} H(h_k) L_k(Z) \; mod \; p$$

$$\text{where, } L_k(Z) = \frac{\prod_{j=1, j\neq k}^{n}((Z) - h_j)}{\prod_{j=1, j\neq k}^{n}(h_k - h_j)} = C_{n-1}Z^{n-1} + C_{n-2}Z^{n-2} + \cdots + C_1 Z + C_0 \; mod \; p \qquad (1)$$

$$C_0, C_1, \dots, C_{n-1} \in Z_p^*.$$

The secret that will be shared amongst all the legitimate attendants is the constant value in (1), $CK = C_0$.

Lagrange polynomial coefficients calculation: We describe now in detail the method we recommend in order the chairperson to calculate the n Lagrange polynomial coefficients C_i of (1) during the *DS* phase. In (1) each basis polynomial $L_k(Z)$ is of degree $n-1$ degree and in canonical form can be written as:

$$L_k(Z) = \frac{\prod_{j=1, j\neq k}^{n}((Z) - h_j)}{\prod_{j=1, j\neq k}^{n}(h_k - h_j)} = c_{n-1}Z^{n-1} + c_{n-2}Z^{n-2} + \cdots + c_1 Z + c_0 \bmod p \; . \qquad (2)$$

It can be easily seen from (2) that:

$$L_k(Z) = \begin{cases} 1, & Z = h_k \\ 0, & Z \neq h_k \end{cases} \qquad (3)$$

Therefore, each polynomial $L_k(Z)$ has exactly $n-1$ roots at the previously computed hash values $h_j,\ j \neq k, j = 1, \dots, n$, namely, $h_1, h_2, \dots, h_{k-1}, h_{k+1}, \dots, h_n$ and hence the coefficients c_i in (2) can be easily calculated by applying Newton's symmetrical functions according to (4):

$$\begin{aligned} c_{n-1} &= 1, \\ c_{n-2} &= -(1 + h_2 + h_3 + \cdots, h_{n-1}), \\ c_{n-3} &= h_1h_2 + h_1h_3 + , \dots, h_1h_{n-1} \\ &\quad + h_2h_3 + h_2h_4 + , \dots, + h_2h_{n-1} \\ &\quad + \cdots + h_{n-2}h_{n-1}, \\ &\dots, \\ c_0 &= (-1)^{n-1}(h_1h_2h_3 \dots h_{n-1}). \end{aligned} \qquad (4)$$

It can be seen that the Lagrange coefficients C_i in (1) can be derived from the coefficients c_i as follows:

$C_i = (\sum_{k=1}^{n} c_{ki} H(h_k)) \bmod p,\ i = 0(1)n-1.$ (5)

Therefore, having derived c_i for each $L_k(Z)$ from (4), the chairperson by applying (5) can calculate the Lagrange polynomial coefficients. For example, in the case that the number of session participants is six, the six coefficients of each fifth-degree polynomial $L_k(Z)$, $k = 1(1)6$ can be calculated by applying (4). Then the Lagrange polynomial coefficients C_i can be derived applying (5).

Next, U_c computes the check value of the shared secret and adds the timestamp T as $V = H(CK \,||\, Z_c \,||\, T)$ before U_c broadcasts the message including all C_i except from C_0.

$$M = (Z_c, V, T, C_{n-1}, C, \dots, C_1) \qquad (6)$$

Note: In order to increase the security of our system and prevent attacker from solving a small-degree Lagrange polynomial (sessions with small membership case) we recommend that U_c generates additional pseudo-random coordinate pairs $\{h_i, H(h_i)\}$ to increase the polynomial degree. However the number of the artificial curve points should be adjustable to the capacity of the nodes since the interpolation computational cost increases $O(n^2)$ with the polynomial degree n or, equivalently, with the number of non-zero coefficients that have to be calculated from (4) and (5).

Share Combiner (SC)

In this phase, each participant U_i in the conference receives the message M and performs the share combiner recovery procedure, where only legitimate $U_i \in B$ (i.e., those users that have already established a pair-wise key with U_c) can recover the correct CK after Step1 to Step4.

Step1. First, U_i verifies the expiration of the received timestamp, T and if it is invalid, U_i terminates the recovery process.

Step2. Second, U_i computes the ECDH pair-wise key with U_c, as $k_{ci} = Z_i Q_c$.

Step3. Third, U_i computes $h_i = H(k_{ci} \,||\, Z_c \,||\, T) \,||\, m$ and solves CK from the following equality:

$$H(h_i) = c_{n-1}(h_i)^{n-1} + c_{n-2}(h_i)^{n-2} + \cdots + c_1 x + c_0 \bmod p$$
$$\xrightarrow{yields} CK = c_0 = H(h_i) - c_{n-1}(h_i)^{n-1} - c_{n-2}(h_i)^{n-2} - \cdots - c_1 x \bmod p \quad (7)$$

Step4. Finally, U_i checks the validity of CK by verifying

$$H(CK \,||\, Z_c \,||\, T) = V. \quad (8)$$

We construct our polynomial without using identities of attendants and we recover the shared secret CK by using polynomial equation system in SC phase. Therefore, our scheme does not require any user identification information or unecessary computation costs for the attending members. The attending members are not allowed to encrypt messages with the session key k_{ci} but only with the CK to avoid chosen plaintext attacks.

V. Performance Analysis

The three tables that follow show the computation, communication and storage cost of the two examined schemes per node in a group of *n* in total nodes (with chairperson). Polynomial computations are done in Z_p^* The main advantages of our proposal is significant storage space saving and computational offload for the chairperson, dynamicity, which is essential for wireless ad hoc networks (see Tables I and III), smaller broadcast message size and no assumption of assigning the computational overhead to the system or delivering keys through private channels..

TABLE I. Number of Operations per Node – PPG Phase

PPG	Operations	U_i		U_c	
		Kim	***Our***	***Kim***	***Our***
Comp.	Random integer generation			*n* for all $\{x_i\}$	2 for Z_C
	EC scalar-point multiplications			*n* for all $\{Q_i\}$	1 for Q_c
	Modular integer multiplications				
	Hashes				
Comm.	Authenticated Broadcasts			1 with Q_c (and all $\{x_i\}$)	1 with Q_c
	Unicasts			-	-
Storage	Message size	$x_i + Q_C$	Q_C	*n* for all $\{Q_i\}$	Q_C

TABLE II. Number of Operationsper Node – DS Phase

DS	Operations/node	U_i		U_c	
		Kim	***Our***	***Kim***	***Our***
Comp.	Random integer generation	-	2 for (Z_i)		
	EC scalar-point multiplications	-	1 for Q_i	*n* for all $\{k_{ci}\}$	*n* for all$\{k_{ci}\}$
	Modular integ. multiplications			*Un-specified*	n^2
	Hashes			2n+1	2n+1
Comm.	Broadcasts			1(M)	1(M)
	Unicasts		1 with Q_i	-	-
Storage	Message size	$x_i + Q_C$	$Z_i + Q_c + Q_i$	*n* for all $\{Q_i\}$	Q_C + all $\{Q_i\}$

TABLE III. Number of Operations – SC Phase

SC	Operations/node	U_i		U_c	
		Kim	***Own***	***Kim***	***Own***
Computations	Random integer generation				
	EC scalar-point multiplications	1 for k_{ic}			
	Modular integer multiplications	*Un-specified*	n-1 *(Horner)*		
	Hashes	3			
Storage	Message size	$x_i + Q_C$	$Z_i + Q_c$ Q_i deleted	*n* $\{Q_i\}$	Q_c

VI. Re-keying And CKDS

The use of ECC is attractive in the case of low-resource networks (e.g., in WSN). Consequently, ECC-based CKDS can also be attractive in generating and distributing a shared symmetric key among the nodes of wireless ad hoc clusters. However, if an ECC-based CKDS scheme is to be utilized into those highly dynamic networks, attention should be paid on efficient re-keying.

Kim et al. do not consider dynamic sessions. Their procedure has to be restarted whenever a new participant is added in a session. On the contrary, in the wireless ad hoc environment we have to consider that mobile nodes join and leave the clusters quite frequently under the same CH. We reduce the computational cost to calculate a new shared conference key from the previously calculated key parameters by proposing two methods.

a) First, by utilizing numerical analysis methods. We propose that if a new node joins (leaves) an ad hoc cluster, provided that its pair of coordinates $\{h_i, H(h_i)\}$ is known to the cluster head, then the latter can calculate a new C_0' from C_0 by applying recursively *Newton's divided difference polynomial formula* [16]. For a single join the new Lagrange interpolation polynomial (1) will be of degree n and a new CK can be recursively derived from the previous polynomial of n-1 degree.

b) Secondly, by exploiting the inherent characteristics of the wireless ad hoc network. To reduce the expensive re-keying calculations, we recommend to utilize the periodic

hello messages which the ad hoc nodes (e.g., sensors) broadcast for building their two-hop Neighboring Lists [15]. The cluster head can utilize the Neighboring Lists to predict which are the distant mobile nodes that have the largest probability to join the cluster. Then he can construct the corresponding additional points $\{h_i, H(h_i)\}$ increasing so the degree of the Lagrange interpolation polynomial. In this way, the CKDS becomes more secure (as usually not without computational cost), and when new nodes will join the cluster, there will be no need to restart key establishment since the new Lagrange polynomial coefficients will have been derived *a priori* due to inclusion of the predicted points in the claculation for the CK.

VII. Security Analysis

We proposed a CKDS scheme that follows well-defined cryptographic assumptions: the intractability of computing the ECDLP, the hardness of inverting a one-way function and the pseudo-randomness of the coordinates. If these assumptions can be solved easily, then CKDS cannot provide user anonymity and data privacy.

In addition, we made adaptations for the dynamic and structured wireless ad hoc network assuming that the legitimate ad hoc nodes have the necessary capacity to generate a public-private key pair using an ECC. Considering that, each node $U_i \in B$ belonging to a cluster dynamically generates an elliptic curve key pair, whose secret key $Z_i \in [1, p-1]$ i.e., $x_i, y_i \in [1, p-1]$ is secured inside the node and the public key Q_i is unicasted to the cluster head node who may be changing. This section presents several basic attack scenarios to demonstrate how the security is enhanced by the proposed CKDS scheme in comparison to the previous mentioned schemes.

Attack scenario 1 (conspiracy attack): Assume that an *intruder* attacker who has recovered CK tries to find the session keys of other legitimate attendants. Knowledge of point Z_i of another attendant would allow to compute his session key k_{ci}. However, in order to find the pseudo-random coordinates of point Z_i, the attacker needs either to solve ECDLP or brute force the [1, p-1] space. In addition, if wanted to reveal k_{ci} attacker should have to launch brute force attack to guess k_{ci} from (unknown) h_i that in addition must belong to the polynomial curve. The same applies when more than one cluster nodes collaborate to reveal the session key of other attendants (ad hoc cluster members). That is more secure than the Yang et al. scheme in which the participant attackers can obtain h_i and try launching brute force attack against the one-way function to obtain k_{ci} and ID_i. Regarding Tseng-Jan scheme, Yang et al. have already shown that it is vulnerable to conspiracy attack.

Attack scenario 2 (eavesdropper): Assume an external attacker who tries to reveal the common share secret CK after capturing the message M broadcasted in DS phase. The attacker should first have to compute the hash value $h_i = H(k_{ci} \,||\, Z_c \,||\, T) \,||\, m$ and then try to recover the CK based on the knowledge of the coefficients included in message M. However, external user has not the ability to generate h_i, because the difficulty involved in generating the coordinates Z_i is based on the ECDLP as is the hardness of computing ECDH key k_{ci}.

Attack scenario 3 (breaking anonymity of session members): Assume attendant (cluster member) $U_j \in B$ tries to find the identity of another session attendant. $U_i \in B$ can easily reconstruct the share secret CK. However, it is infeasible to find the identity of another neighboring node since the node identities are not included at any stage of the proposed scheme. In previous CKDS schemes the identity can be revealed with brute force attack in the one-way function $H(\,)$.

Attack scenario 4 (impersonation attack-pretending the chairperson):

An *intruder* tries to replay an intercepted message $M = (Z_c, V, T, C_{n-1}, C_{n-2}, \dots, C_1)$ to impersonate the cluster head U_c and compromise the ad hoc network operation. The attacker should set a new acceptable timestamp T so that the cluster nodes can verify the validity of T in DS phase. Then, the cluster nodes compute k_{ci} and h_i to solve the CK and ckeck the validity of CK by verifying $H(CK \,||\, Z_c \,||\, T) = V$. However, the attacker can not forge a valid CK without knowing Z_c from Q_c. To obtain Z_c from Q_c is equivalent to solving the ECDLP. The cluster nodes can verify the validity of V at SC recovery stage. Therefore, an attacker cannot obtain any secret by replaying an intercepted message M.

VIII. Conclusion

We reviewed previous CKDS schemes and we adapted the Lagrange polynomial interpolation method combined with elliptic curve cryptographic techniques in distributing secret shares in the restricted ad hoc network. We analyzed our scheme with attack scenarios which prove that it overcomes the vulnerabilities of the previously proposed schemes. In the future we are planning to evaluate and optimize the effectiveness and efficiency of our proposal through network simulation and comparisons with other CKDS schemes.

References

[1] I. Ingemarsson, D.T. Tang, and C.K. Wong, "A conference key distribution system", IEEE Transactions on Information Theory IT-28, 1982, pp. 714–720.

[2] T.C. Wu, "Conference key distribution system with user anonymity based on algebraic approach", IEE Proceedings. Computer Digital Technology 144 (2), 1997, pp. 145–148.

[3] Y.M. Tseng, J.K. Jan, "Anonymous conference key distribution systems based on discrete logarithm problem", Computer Communications 22 (1999) pp. 749–754.

[4] C.C. Yang, T.Y. Chang, M.S. Hwang, "A new anonymous conference key distribution system based on the Elliptic Curve Discrete Logarithm Problem", Computer Standards and Interfaces, 25 (2003) pp: 141-145.

[5] C.H. Lin, C.Y. Lee, W. Lee, "Comments on the Yang-Chang- Hwang anonymous conference key distribution system", Computer Standards and Interfaces, 26 (2004) pp: 171–174.

[6] W.H. Kim, E.K. Ryu, J.Y. Im, K.Y. Yoo, "New conference key agreement protocol with user anonymity", Computer Standards & Interfaces, 27 (2005); 185–190.

[7] Q. Tang and C. J. Mitchell, "Comments on two anonymous conference key distribution schemes", Computer Standards & Interfaces, 27 (2005); 397-400.

[8] W.G. Tzeng, Z.J. Tzeng, "Round-efficient conference key agreement protocols with provable security", Advances in Cryptology, ASIACRYPT 2000, Volume 1976 of the series Lecture Notes in Computer Science, pp. 614-627.

[9] W.G. Tzeng, “A secure fault-tolerant conference key agreement protocol”, IEEE Transactions on Computers 51 (4) (2002) 373–379.
[10] K.H. Huang, Y.F. Chung, H. H Lee, F. Lai, T.S. Chen, “A conference key agreement protocol with fault tolerant capability”, Computer Standards & Interfaces; 31 (2009); 401-405;
[11] W.G. Tzeng, “A Practical and Secure Fault-Tolerant Conference-Key Agreement Protocol”, Public Key Cryptography, Volume 1751 of the series Lecture Notes in Computer Science, pp. 1-13.
[12] Y.M. Tseng, “A communication-efficient and fault-tolerant conference-key agreement protocol with forward secrecy”, Journal of Systems and Software, Volume 80, Issue 7, July 2007, pp. 1091–1101.
[13] Y.M. Tseng “An Improved Conference-Key Agreement Protocol with Forward Secrecy”, Informatica 16(2), 275-284 (2005).
[14] N. Komninos, C. Douligeris, “LIDF: Layered intrusion detection framework for ad-hoc networks”, Journal of Ad Hoc Networks, Volume 7, Issue 1, January 2009, pp. 171-182.
[15] C. Tselikis, S. Mitrpoulos, C. Douligeris, N. Komninos, “Degree-based Clustering Algorithms for Wireless ad hoc Networks under Attack", IEEE Communications Letters, Volume 16, Number 5, 2012, pp. 619-621.
[16] R. L. Burden, J. D. Faires, A. Burden, Numerical Analysis, 10th Edition, ISBN-10: 1305253663.
[17] N. Komninos, C. Tselikis, C. Douligeris, “SAnoVS: Secure Anonymous Voting Scheme for Clustered ad hoc Networks”, 8th IEEE Symposium on Computers and Communication (ISCC’13), 07 July 2013, Croatia, pp: 192-196.