Received: 19 May 2023 | *Revised: 27 July 2023* | *Accepted: 27 July 2023*
DOI: 10.37917/ijeee.20.1.21 *Vol. 20 | Issue 1 | June 2024*

Open Access

Iraqi Journal for Electrical and Electronic Engineering
Review Article

Group Key Management Protocols for Non-Network: A Survey

Rituraj Jain*, Dr. Manish Varshney
Department of Computer Science, Maharishi University of Information Technology, Lucknow, U.P. India

Correspondance
*Rituraj Jain
Department of Computer Science,
Maharishi University of Information Technology,
Lucknow, U.P. India
Email: jainrituraj@yahoo.com

Abstract
The phenomenal rise of the Internet in recent years, as well as the expansion of capacity in today's networks, have provided both inspiration and incentive for the development of new services that combine phone, video, and text "over IP." Although unicast communications have been prevalent in the past, there is an increasing demand for multicast communications from both Internet Service Providers (ISPs) and content or media providers and distributors. Indeed, multicasting is increasingly being used as a green verbal exchange mechanism for institution-oriented programmers on the Internet, such as video conferencing, interactive college games, video on demand (VoD), TV over the Internet, e-learning, software programme updates, database replication, and broadcasting inventory charges. However, the lack of security within the multicast verbal exchange model prevents the effective and large-scale adoption of such important company multi-celebration activities. This situation prompted a slew of research projects that addressed a variety of issues related to multicast security, including confidentiality, authentication, watermarking, and access control. These issues should be viewed within the context of the safety regulations that work in the specific conditions. For example, in a public inventory charge broadcast, while identification is a vital necessity, secrecy is not. In contrast, video-convention programme requires both identification and confidentiality. This study gives a complete examination and comparison of the issues of group key management. Both network-dependent and network-independent approaches are used. The study also addresses the advantages, disadvantages, and security problems of various protocols.

Keywords
Communication, Group Key Management. Multicasting, Network, Protocols.

I. Introduction

The relevance of institution dialogue involving more than nodes can be fully grasped from continuous real-time applications like as email, Skype, chat, Facebook, Twitter, and online games, among others. While enterprise communication has seen rapid growth in today's networking environment, security remains a significant challenge. Aside from social networks, more secure environments, such as a naval network, where sensitive information is transmitted, necessitate a personal and consistent environment for information transmission, club control, and key control. As a result, the security of institution discussion is dependent on the secrecy and energy of the institution key employed. Initialization, generation, registration, backup, update, recovery and revocation are the main phases of key lifecycle. Key management is more crucial in group-based communication in non-network where group management has to be managed with key management. It is one of the major concerned with objectives of maintaining the integrity of communicated messages between the group members of non-network.

The key status quo in a group could be key settlement and key distribution. Every cooperating node contribution is required to generate a key and ensure that it is newly generated. In a key distribution strategy, one collaborating node is responsible for producing and distributing the critical thing to all participating nodes of the institution dialogue. Another crucial feature is a rekeying procedure for when the club changes within the dynamic environment.

The institution key control protocols depicted in Fig. 1 are normally classified as Network impartial mostly based totally and Network established largely based totally [1–4]. The community impartial primarily based totally key control protocol is further classified as centralized, decentralized, and distributed key control protocols, whereas the community established primarily based totally key control is classified as tree-based totally and cluster-based totally key control [5, 6].

The role of key control also includes contributing member authentication to prevent intruders from impersonating one and providing access management to authenticate the joining procedure. Furthermore, the key control employs a variety of cryptographic procedures in manufacturing and distributing keys that might be symmetric or unequal for consistent institution communication.

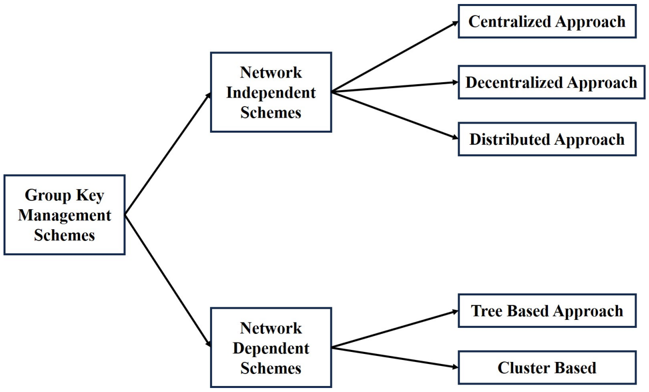

Fig. 1. Classification of Group Key Management (GKM) Schemes [1–6]

II. Necessity of Group Key Management

Confidentiality, integrity, and authenticity are the fundamental security criteria. The main requirements can be divided into three categories: performance requirements, efficiency requirements, and safety requirements [7–9]. Efficient group key management techniques must consider multiple criteria such as security, service quality, key server resources, and group member resources [10].

A. Security Requirements

1. According to the Forward Secret, individuals who have left the group must not have access to keys in the future. Because of this, a member cannot decrypt data after leaving the group. The best way to guarantee forward secrecy is to re-encrypt the pool with a new TEK upon each exit from the pool [8].

2. In order to maintain reverse secrecy, a new user that joins the session must not have access to the old keys. This makes sure that someone who joins the group later cannot decrypt the data being sent. The best option is to encrypt the group with a new TEK after each group join in order to guarantee backward secrecy.

3. The absence of collusion necessitates that no rogue user group can obtain the encryption key from legitimate traffic.

4. Key Independence: A protocol is deemed key independent if the revelation of one key has no impact on the security of any other keys [11–13].

5. Minimum trust: A large number of entities should not have confidence in the key management system. Otherwise, it would be difficult to deploy the system effectively.

B. QoS Requirement

1. Low message overhead: Rekeying groups should not consume a lot of bandwidth, especially for dynamic groups. Idealistically, this ought to be unrelated to group size.

2. 1-affects-n: A procedure exhibits the 1-affects-n phenomenon if a single change in the membership of the group has an impact on every other member. This usually happens when a single membership change necessitates the adoption of a new TEK by the entire group.

3. Minimal Delays: Many multicast-dependent applications (usually multimedia applications) are sensitive to packet delivery jitter and delays.
Any key management system should, therefore, take this into account and try to reduce how key management affects packet delivery delays.

C. Group Member Resources

1. Limited storage space refers to the small number of keys required for communication, enabling efficient operation of key servers and speedy access from memory. Greater storage may necessitate more processing power and memory for key management [14, 15].

2. Low computing power: The computation was low if the group members and key server both use the minimum number of keys. Improved efficacy and responsiveness of key servers to group members is a benefit of less commute [16, 17].

III. Network Independent Schemes

Any user connected to a wireless network can access the delivered packets because it uses a shared media. Through the use of encryption techniques, access can be restricted to the group [18]. Therefore, to prevent unauthorized users from viewing the data, it might be encrypted utilizing shared key communication. The shared encryption key (TEK), also known as the traffic encryption key or group key. The safety of group communication completely depends on this, and it is the key [19, 20].

A. Centralized Group Key Management Schemes

In a centralized system, group communication is handled by a single entity. This entity is responsible for key creation, distribution, and management. The following are the major disadvantages of a centralized scheme:

Because the efficacy of group communication is based on a single centralized organization, rekeying becomes an administrative burden as the group size increases. The number of keys required to be stored for a session. There is just one failure point. When a new member joins the club or an existing member leaf, the difficulty of maintaining forward and backward secrecy arises. Expelled members must work together and disclose their own information in order to reclaim access to the group key.

In wireless contexts, centralized group key management is used. To facilitate key management during key distribution and updating, schemes such as LKH [21] and OFT [22] use a single KDC (Key Distribution Centre) with a hierarchical key structure. In a large and highly dynamic wireless group application, frequent rekeying may overload the capability of a single KDC, resulting in key management activities failing [11]. This failure would threaten the group application's security. Additionally, as the number of group members increases, members must handle an increasing number of rekeying messages. The frequent keying required by a big group with changing membership may exceed the capabilities of lightweight mobile devices. The first issue encountered is the inability of centralized systems in wireless networks to cope with rising group size (i.e., lack of scalability). Centralized group key management systems are the most well-known and commonly utilized schemes [23–27]. The centralized key management strategy is further classified as paired key technique [21, 22], key centre group identity, making a group key, group key distribution, key regeneration, secure lock approach, keys hierarchy approach, and one way function trees.

The pairwise key method: The single point entity in this approach shares pairwise keys with each group member. The Group Key Management Protocol (GKMP) is an example of this method. GKMP is a mechanism described in [21, 22] that combines pairwise key generation with key distribution techniques from a Key Distribution Centre (KDC) to distribute a symmetric key to a member of a multicast group.

The Key Centre Group's identity: The initiator of the multicast group acquires a group management certificate from the certification hierarchy. The certificate holder is solely responsible for the group key's development and dissemination.

Making a Group Key: The Group Key Management (GKM) program, when contacted, picks a group member and generates a Group Key Packet (GKP) on behalf of the certificate holder. GKP holds the paired key (the current Group Traffic Encrypting Key (GTEK) and a future Group Key Encrypting Key (GKEK).

Key Distribution in a Group: The GKM contacts each group member and validates and generates a group session key (session TEK and KEK) and group rekey (EKEK (GKP)), both of which are signed with the originator's certificate.

Regeneration of Keys: When a new member joins the group, the GKM program acts as an originator and generates a new GKP and a new group rekey (encrypted with GKEK), which are then broadcast to the group members.

Secure Lock Approach: When a member exits in a single broadcast, a single point entity establishes a group key or a rekey operation, according to the researchers in [28]. The central entity computes the Chinese Remainder for each message before sending it to a group member, however the number of rekey messages is greatly reduced.

Keys Hierarchy Approach: The central entity reduces the overhead associated with rekeying by distributing the secret keys with subgroups of the entire secure group. As a result, when a member leaves the current session, the central entity distributes the new TEK using the secret keys provided with subgroups because the departing member is unaware of the shared secret keys. This approach also uses pairwise keys and trades off storage overhead for reduced rekeying overhead. The hierarchy technique is used by protocols such as Logical Key Hierarchy (LKH), One Way Function Trees (OFT), and Centralized Flat Table Key Management (CFKM).

In LKH [29–31], the TEK is held by the tree's root. The nodes in the tree's interior layers, known as KEKs, hold keys along the path from a leaf to itself, whilst the leaves of the tree hold the secret keys shared by the group members. A balanced binary tree member can only have 1 + log2n keys, where n is the number of group members. Fig. 2 depicts the key hierarchy in LKS. OFT is a LKH extension project. The number of rekey messages is reduced by half to log2n. In OFT [32, 33], group members compute KEKs, whereas the root entity is in charge in LKH. As a result, when compared to the LKH approach, the number of messages required for rekeying is cut in half with this strategy. OFT is related to another approach, one way Function Chain tree [34]. This

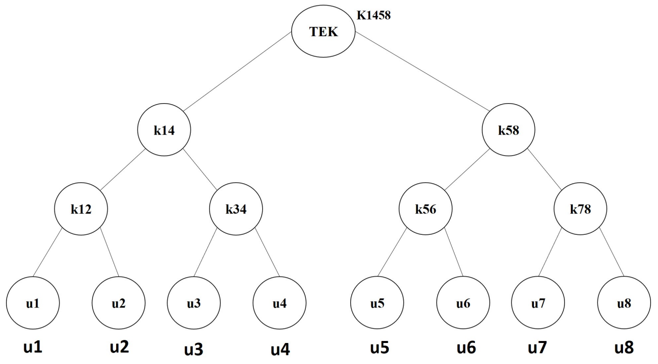

Fig. 2. Hierarchy of Keys in LKH [21]

technique works similarly to OFT, except instead of a one-way function, it uses a pseudo random generator to generate new KEKs. The CFKM approach introduced in [35] saves keys in flat tables to lower the cost of key management by the central entity. The table entries contain one TEK and two KEK entries, where w is the number of bits in the member id (preferably IP address).

TABLE I shows the efficiency of the Centralized key Management strategies discussed thus far. In TABLE I, Kb is the key size in bits, h is the tree height, Id is the number of bits in the member ID, and gm is the number of group members.

B. Decentralized Group Key Management Schemes

Decentralized architecture addresses scalability for group key management in a vast area by separating the overall group into multiple tiny subgroups. These techniques are appropriate for managing group keys in large-scale wireless networks such as cellular networks, WiMax [36,37], and future 4G systems [38]. Furthermore, the only method to address the 1-affect-N issue, which must also be considered when dealing with wireless networks, is through decentralized architecture. In contrast, decentralized architecture simply provides a framework for large-scale group key management; it does not propose a way for efficiently distributing keying materials to group members in subgroups. As a result, decentralized architecture must interact with other group key management methodologies in order to provide an integrated solution for group key management in wireless networks. Furthermore, third-party entities are typically involved in wireless network decentralized systems. Wireless network operators and secure group application providers are often distinct businesses. Trust must be established between them, which is a crucial security risk that wireless group key management techniques must handle as well. The protocols based on decentralized GKM techniques include intra-domain group key management (IGKMP), Scalable Multicast Key Distribution (SMKD), Dual-Encryption Protocol (DEP), Hydra, Kronos, and Marks Protocols. The IGKMP architecture suggested in [39,40] divides the network into administratively scoped regions, with a Domain Key Distributor (DKD) and an Area Key Distributor (AKD) for each accessible area. The DKD is in charge of developing the group's key TEK and disseminating it to members via AKD.

The IGKMP architecture is depicted in Fig. 3. DKD and AKD are assigned to the all-KD-group multicast group, where DKD keeps watch and AKD keeps track of members.

Failure of DKD results in a breakdown in overall group communication, whereas failure of AKD results in a breakdown in particular area communication.

The SMKD suggested in [41] uses the Core Based Tree (CBT) multicast routing protocol to build a multicast tree. A primary central entity and secondary entity cores make up the CBT. The primary core entity produces an Access Control List (ACL), a group session key (GTEK), and a key encryption key (GKEK) in order to update GTEK. These keys are given to secondary cores and other nodes when they join the multicast tree once the joining nodes have been authenticated. The central core is the only entity that generates session keys; the secondary core is authenticated by the primary core, which in turn authenticates the joining members and distributes the keys via the ACL. Forward secrecy in SKMD is still a problem that has not been solved. The issue of trusting third parties is resolved by this decentralized time driven protocol because there are many available intermediary nodes in the decentralized method.

Members of the DEP [42] subgroups are organized hierarchically, with a Sub-Group Manager (SGM) in charge of each subgroup. In this instance, there are three KEKs and one DEK (Date Encryption Key). A SGMi and the individuals in its subgroup have KEKi1. The members of subgroup I, with the exception of SGMi, and the Group Controller (GC), share KEKi2. Finally, SGMi and GC share KEKi3. By first encrypting the DEK produced by the GC with KEKi2, and then encrypting it once more with KEKi3, the DEK is transmitted to the group members and the GC. Before transmitting the encrypted DEK packet to subgroup i, SGMi first decrypts it with KEKi3 and then re-encrypts it with KEKi1, which is shared by the members of the subgroup. Now, each member of subgroup I decrypts the message using KEKi1 and KEKi2, recovering the DEK along the way. Members who own both keys are the only ones who can access DEK. Because they are unable to access DEK because they are unfamiliar with KEKi2, SGMs end up being a reliable third party. Every time a member of the subgroup i enters or leaves, SGMi updates KEKi1 and transmits it to the members of the subgroup i.

Due to modifications in DEK, subgroup I members who have not yet received the new KEKi1 will not be able to access resources. If DEK keeps the same, a forward secrecy issue arises since members who did not get KEKi1 can still access

TABLE I.
SUMMARIZATION OF CENTRALIZED GKM SCHEME'S COMPARISON

Protocol Source		[21, 22]	[29, 30]	[32, 33]	[35]
1-affects-n		✓	✓	✓	✓
FS[a]		X	✓	✓	✓
BS[b]		✓	✓	✓	✓
CF[c]		✓	✓	✓	X
OR[d]	Joining	2K	(2h–1)Kb	(h+1)Kb	2IdKb
	Leaving	-	Id + 2hKb	Id+(h+1)Kb	2IdKb
	Overhead Status	✓	-	-	-
OS[e]	KDC	2K	(2gm–1)Kb	(2gm–1)Kb	(2Id+1)Kb
	Member	2K	(h+1)Kb	(h+1)Kb	(Id+1)Kb
	Overhead Status	-	✓	✓	✓

a. Forward Secrecy b. Backward Secrecy c. Collision from Freedom d. Overhead in Rekey e. Overhead in Storage

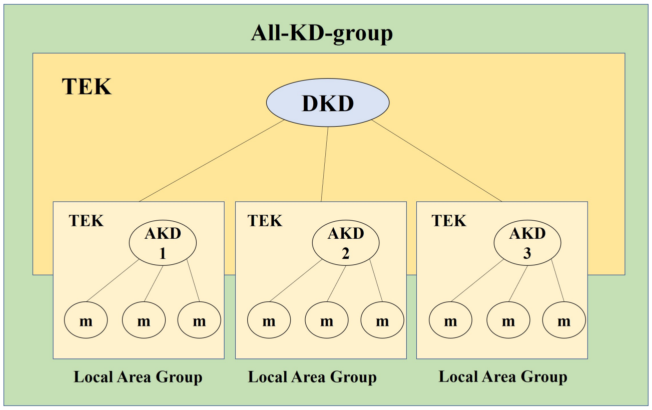

Fig. 3. Architecture of Intra-Domain Group Key Management

the multicast session.

The Hydra protocol [43] ensures that fresh group keys are generated by a single trusted HS whenever a departure or joining action takes place by adopting a decentralized group key management mechanism to divide the session group into smaller subgroups and send the group key to all HS via SGKDP. Rekeying happens at regular periods of time using the time-driven Kronos technique, regardless of whether people join or leave the group. The architecture of Kronos, as it is described in [44], is similar to IGKMP. In contrast to IGKMP, AKD is in responsible of generating the group key and distributing it on a regular basis to members within its zone. All AKD clocks are synchronized so that all AKDs agree on the rekeying time period in order to have the same group key communicated after a period of time. For clock synchronization (NTP), the Network Time Protocol is suggested. Apart from synchronization, all AKDs must agree on two secret factors, R0 (an initial value) and K, the master key, both of which are supplied by DKD via a secure channel. These hidden elements are used by AKDs to generate subsequent keys. MARKS were suggested in [45]. In this time-driven technique, the time slices at which rekeying is performed are safeguarded by encrypting each time slice with a unique key. The encryption keys utilized in the architecture are generated from a single seed and serve as leaf nodes in a binary hash tree. Internal nodes are also referred to as seeds.

TABLE II illustrates the performance of the membership-driven approach and time-driven approach protocols of a decentralized group key management scheme. Key independence, subgroup rekeying, and central rekeying are among the factors used to compare the protocols. Aside from these characteristics, protocols are labeled as fault tolerant or not.

TABLE II.
SUMMARIZES THE PERFORMANCE OF DECENTRALIZED GKM SCHEME'S COMPARISON

Protocol Source	KI[a]	1-affects-n	LR[b]	Rekey	FT[c]
[39, 40]	✓	✓	X	✓	X
[41]	✓	✓	X	X	X
[42]	✓	✓	X	X	X
[43]	✓	✓	X	✓	✓
[44]	X	X	X	X	✓
[45]	X	X	X	X	✓

a. Key Independence b. Local Rekey c. Fault Tolerance

C. Distributed Group Key Management Schemes

The distributed GKM approach requires the multicast session's group members to collaborate in order to generate the required group key. Because there is no group controller involved, this strategy makes the system failure-tolerant. However, when the group membership changes, the distributed key management scheme compromises the security mechanisms; second, as the group size grows, processing time and communication overhead increase; and third, in order to ensure reliable communication, each member must keep track of the other members participating in the multicast session.

This key management strategy is further classified into three types based on the virtual topology created by the working group members: ring-based collaboration, hierarchy-based cooperation, and broadcast-based cooperation.

The following variables affect the dispersed approach:

- The quantity of rounds necessary for processing and communication.
- The quantity of messages that will be sent and received by the group members.
- Group key is generated computationally.

List of protocols based on the ring, hierarchy and broadcast cooperation in distributed GKM scheme is given Fig. 4.

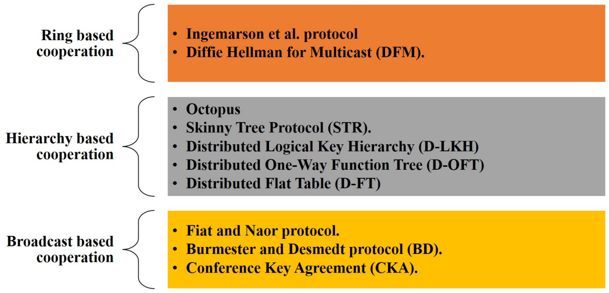
Ring based cooperation
• Ingemarson et al. protocol
• Diffie Hellman for Multicast (DFM).
Hierarchy based cooperation
• Octopus
• Skinny Tree Protocol (STR).
• Distributed Logical Key Hierarchy (D-LKH)
• Distributed One-Way Function Tree (D-OFT)
• Distributed Flat Table (D-FT)
Broadcast based cooperation
• Fiat and Naor protocol.
• Burmester and Desmedt protocol (BD).
• Conference Key Agreement (CKA).

Fig. 4. Categorization of Distributed GKM Scheme

Ring-based group key management players form a virtual ring. The protocol described by Ingemarson et al. in [46] is an example of a protocol based on this category. This protocol is an adaptation of the Diffie Hellman key agreement mechanism for group communication. The group's members create a virtual ring. When the group membership changes, the entire process must be performed to generate the new group key. The DFM protocol [47] defines a two-party Diffie Hellman key exchange protocol [48] to n-party communication extension. Each member of the group agrees on two primes. Each player chooses their own secret key. Initially, the first member computes its key and passes it to the second member, who computes its key and passes the combined key to the third member, who repeats the process until it reaches the last member. The final member broadcasts the final key to the entire group after computing it. Each member extracts their own intermediate value and generates k when acquiring the group key.

The Octopus protocol [49] is an enhancement to the Diffie Hellman Key (DHK) exchange mechanism. The multicast group is divided into subgroups of four members each. Each subgroup determines and computes the intermediate key subgroup value, which it then shares with the other subgroups. Each subgroup's leader is in charge of exchanging the intermediate key I. A tree structure is used in the STR technique, which was devised in [50]. Because the tree is arranged linearly, determining the group key takes O(n) time. Each tree member is in charge of storing and preserving all public keys connected with the tree's nodes. When a member joins or leaves, the tree is rebuilt, and all members change the group key to create a new key kn that is linked to the tree's root. The D-LKH protocol has no GC [51]. The generated hierarchy is divided into two subtrees, left and right, with row subtree members agreeing on a mutual group key for encryption.

D-OFT [52] is a centralized approach that does not have a group controller. Every member of the team is in charge of access control and key creation. Each member creates their own key and sends a blindfolded version to their sister. D-FT [28] members only know their KEKs. The distributed technique has the disadvantage of requiring a new member to contact a group of members in order to get all of the necessary keys. Because numerous members may be changing the same key at the same time, key synchronization may be delayed.

The Fiat and Naor protocols are based on Diffie Hellman property [53]. A trusted and dependable center T initiates the system under this protocol. T chooses two prime numbers, p and q, and broadcasts the formula n = p.q to all nodes. T then generates and conceals a random number g. T transmits two random xi values and a key i to new member Mi when she joins the group. To get an agreement on a group key K, each member broadcasts their xi values and therefore computes K.

CKA is proposed in [54] as a distributed technique in which all members of the group contribute to the production of the group key. K = f(N1, h(N2),..., h(Nn)), where f is the combining function (a MAC), h is a one-way function, n is the number of group members, and Ni is the contribution from group member i. The protocol requires n1 members to make their contributions public (Ni). The group leader, for example, U1, encrypts and transmits its contribution (N1) using each member's public key. Everyone in the group who has the public key can decrypt and generate the group key.

The Burmester and Desmedt protocol [55] is a three-round distributed key management protocol. The member mi randomly generates ri and broadcasts Zi. Member mi computes and broadcasts Xi. The key Kn is determined by the member mi. This protocol requires n+1 exponentiations per member, with the exponent in all but one case being at most n1. This protocol has the problem of requiring 2n broadcast messages.

Distributed key exchange scheme requirements are described in TABLE III. The number of rounds, the number of multicast messages required, the Diffie Hellman keys, and the number of required leaders are all compared. Because all group members contribute in creating the required group key under the distributed approach, every member must do the required computations again when a member joins or leaves the group, this technique addresses the 1-affects-n problem.

TABLE III.
SUMMARIZE THE PERFORMANCE OF DISTRIBUTED GKM SCHEME'S COMPARISON

Protocol Source	NC[a]	MM[b]	DH-K[c]	LR[d]
[28]	Y	-	X	✓
[46]	m-1	-	✓	X
[47]	Y	m	✓	X
[49]	2(m-1)/4+2	-	✓	✓
[50]	Y	m	✓	X
[51]	3	1	X	✓
[52]	$\log_2 n$	-	X	X
[53]	2	m	✓	✓
[54]	3	m	X	✓
[55]	3	2m	X	X

a. Number of Cycles b. Multicast Messages c. DH-Key d. Leader Requirements

IV. NETWORK INDEPENDENT SCHEMES

Wireless networks cannot support network-independent group key management protocols. To execute efficient group key management across wireless networks, group key management protocols should be network dependent. To be implemented or run efficiently, such group key management methods must rely on elements of the underlying network architecture. One of the biggest issues of network-dependent group key management protocols is supporting mobile multicast, in which members travel across the wireless network while continuing to receive their subscribed multicast services.

To receive multicast services efficiently, members must be moved from one location to another throughout the network, adding complexity to key management and traffic control.

The difficulty of key handling emerges whenever a member quits or joins the group, increasing the group's complexity both intra and interdomain, so that data transmission security is maintained while overall system performance is unaffected. The methods listed for mobile multicast security group key management are classified into two types: tree-based and cluster-based techniques. Tree-based approaches include Topology Matching Key Management Tree (TMKM) [56, 57], A Hybrid Key Management Scheme (HKM) [58], and WANG Approach [59, 60], whereas cluster-based approaches include Gharout et al. protocol [61,62], Kellil et al. protocol [63], and Group Key Management Framework (GKMF) protocol [64].

A. Tree Based Approach

The TMKM [56,57] protocol implements a LKH key tree with a three-level topological structure. The network is divided into three components: mobile users, Base Stations (BS), and a Supervisor Host (SH). The SH-controlled BSs handle keys within their cell and broadcast group key information to their members. The SH handles mobile user routing and creates the necessary keys for secure group communication, such as the group key (TEK) and supporting keys. When members move across cells, an efficient handoff mechanism relocates the user in the TMKM tree. HKM is a wireless environment concept by [58] that is similar to TMKM and includes Topology Matching (TM) and Topology Independent (TI) sub-trees. The HKM tree handles both high mobility and low mobility members by combining the capabilities of two key management trees, TIKM trees and TMKM trees, and eliminating rekeying overheads during the handoff process. Thanks to these methods, which allow rekeying messages for low mobility users to be relayed to the designated location, rekeying messages for high mobility users only need to be broadcast when the users leave the group, regardless of the number of handoffs taking place. The high mobility users are classified into the TI sub tree of the HKM tree, whereas the low mobility users are divided into the TM sub tree. High and poor mobility are assessed based on the member's movement velocity. In [59, 60] present a distributed network-dependent group key management system with group members divided into leader units and general member units. The leader units are in charge of the primary management. This protocol also manages null area re-keying when members switch cells. This technique employs a handoff member mechanism to deal with member mobility. The mechanism is made up of a two-tier logical architecture that correlates to the cellular network structure. In this two-tier system, the key server significantly reduces the communication overhead that occurs during key updating. The network entities used (CKS) are Group Key Servers (GKS) and Independent Cell Key Servers (ICKS).

B. Cluster Based Approach

In [61, 62] a novel key management protocol with the goal of providing secure group communication in a mobile network environment while requiring no rekeying. The technique uses independent TEK for each subgroup, which overcomes the 1-affect-n phenomena. The network realities are the Domain Key Distributors (DKDs), which oversee all of the Area Key Distributors (AKDs) under its control, and AKDs, which execute critical operations for their area as well as authenticate mobile members. The sharing members are divided into clusters, with each cluster regulated by one DKD on the sphere and at least one AKD on the region. No rekeying is required when a member transfers inside the same cluster, which improves the rekeying procedure for AKDs under the same DKD. This approach guarantees forward and reverse secrecy services. Mobile members participating in several sessions must store multiple encryption keys, resulting in storehouse outflow. This solution does not address the complexities of rekeying when members transfer between clusters.

Decentralized area re-keying techniques for mobile multicast communication that address member mobility are suggested in [63]. This technique reduces the overhead of area rekeying by using two unique rekeying mechanisms for static and mobile members. The ability to move between locations ensures backward and forward secrecy. The protocol suffers from the 1-affects-n phenomena and is unable to handle highly dynamic and mobile members due to numerous rekey requests because it uses a common TEK method.

The primary purpose of GKMF [64] is to provide Secure Group Communication in Wireless Mobile Environments. The protocol employs lists to handle the dynamic members of the cellular network environment. There are two kinds of network entities: primary entities and placement entities. Based on these entities, the participating members or nodes are divided into two levels: domain level and area level. DKM (Domain Key Manager) entities generate, distribute, store, and remove all key material required at the domain level. The entity at the area level is the Area Key Manager. It supervises the members of its group and is in charge of key management in its domain. GKMF uses shared symmetric keys to construct safe associations at several levels, ensuring a reliable link between communicating entities. The protocol's opposing side suffers from storage overhead because a large number of utilized keys must be kept. Because it does not handle the re-keying operation in the area from which the member is exiting, the protocol does not ensure forward secrecy and suffers from 1-affect-n owing to a common TEK. Because area and TEK rekeying are conducted separately, the protocol provides backward secrecy but introduces a joining delay.

TABLE IV analyzes network-dependent protocols based on factors such as key dependency, the 1-affects-n phenomenon, handling multiple membership changes, scalability, security service support, fault tolerance, and rekey overhead. The Wang technique, as compared to other protocols under consideration, may operate in heterogeneous wireless networks. The centralized nature of the protocol (TMKM, HKM) results in a single point of failure, and the number of tree levels in the protocol structure also has a substantial influence on performance when more handoffs carried over. There is a single point of failure in the protocols outlined in [63] and GKMF.

TABLE IV.
SUMMARIZATION OF THE PERFORMANCE OF A NETWORK DEPENDENT GKM SCHEME'S COMPARISON

Scheme	Protocol Source					
	[56, 57]	[58]	[59, 60]	[61, 62]	[63]	[64]
TA[a] /CA[b]	TA	TA	TA	CA	CA	CA
KI[c]	✓	✓	✓	✓	✓	✓
1-affects-n	✓	✓	✓	✓	✓	✓
MMCS[d]	X	X	✓	X	X	X
S[e]	✓	✓	✓	✓	X	X
SS[f]	X	X	✓	✓	X	X
FT[g]	X	X	✓	✓	X	X
OR[h]	✓	X	✓	X	✓	✓

a. Tree based Approach b. Cluster Based Approach c. Key Independence d. Multiple Membership Change Support Requirements e. Scalability f. Security Service g. Fault Tolerance h. Overhead of Rekey

V. CONCLUSION

The study explores various ways to group key management in both network and non-network situations. The poll clearly demonstrates that each protocol has distinct characteristics, regardless of whether it employs a centralized, decentralized, or distributed structure. The centralized strategy is easy to implement. Scalability is made possible via the decentralized framework. The distributed framework structure allows every participant to participate in key management activities by splitting the members of the participating group into subgroups. The security of the TEK used is critical to the success of multicast communication. As a result, effective group key management is required to produce, distribute, and update the group key in a secure manner over an unprotected connection. The survey addresses the use of a common TEK method and an independent TEK approach for each subgroup. To propose an effective key management system, variables such as delay, the 1-affect-n phenomenon, storage overhead, and rekey overhead must be properly considered. The resource constraints, bandwidth constraints, highly dynamic environment, and rapidly changing membership must all be considered for effective key management and successful multicast communication in the cellular network environment. The study conducted a meta-analysis on GKMP for non-networks such as MANETs or VANETs to demonstrate the effectiveness of security algorithms in non-networks. The study also highlighted new contributions and research activities, along with crucial features that will guide future researchers in designing safe, secure, and efficient non-network communication models.

Conflict of Interest

The authors have declared no conflict-of-interest statements.

References

[1] R. Nasri and A. Bourouis, “Evaluation of cryptographic key management systems in wireless ad-hoc networks,” 2022.

[2] K. V. Kumar, T. Jayasankar, V. Eswaramoorthy, and V. Nivedhitha, “Sdarp: Security based data aware routing protocol for ad hoc sensor networks,” *International Journal of Intelligent Networks*, vol. 1, pp. 36–42, 2020.

[3] M. G. El-Hadidi and M. A. Azer, “Traffic analysis for real time applications and its effect on qos in manets,” in *2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, pp. 155–160, IEEE, 2021.

[4] W.-C. Wu, H.-T. Liaw, *et al.*, “A study on high secure and efficient manet routing scheme,” *Journal of Sensors*, vol. 2015, 2015.

[5] V. S. Devi and N. P. Hegde, “Multipath security aware routing protocol for manet based on trust enhanced cluster mechanism for lossless multimedia data transfer,” *Wireless Personal Communications*, vol. 100, pp. 923–940, 2018.

[6] V. Gomathy, N. Padhy, D. Samanta, M. Sivaram, V. Jain, and I. S. Amiri, “Malicious node detection using heterogeneous cluster based secure routing protocol (hcbs) in wireless adhoc sensor networks,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 4995–5001, 2020.

[7] R. Kousar, M. Alhaisoni, S. A. Akhtar, N. Shah, A. Qamar, and A. Karim, “A secure data dissemination in a dht-based routing paradigm for wireless ad hoc network,” *Wireless Communications and Mobile Computing*, vol. 2020, pp. 1–32, 2020.

[8] M. Maheswari, S. Geetha, S. S. Kumar, M. Karuppiah, D. Samanta, and Y. Park, “Pevrm: probabilistic evolution based version recommendation model for mobile applications,” *IEEE Access*, vol. 9, pp. 20819–20827, 2021.

[9] L. E. Funderburg and I.-Y. Lee, “A privacy-preserving key management scheme with support for sybil attack detection in vanets,” *Sensors*, vol. 21, no. 4, p. 1063, 2021.

[10] A. Hammamouche, M. Omar, N. Djebari, and A. Tari, “Lightweight reputation-based approach against simple and cooperative black-hole attacks for manet,” *Journal of information security and applications*, vol. 43, pp. 12–20, 2018.

[11] N. Veeraiah, O. I. Khalaf, C. Prasad, Y. Alotaibi, A. Alsufyani, S. A. Alghamdi, and N. Alsufyani, “Trust aware secure energy efficient hybrid protocol for manet,” *IEEE Access*, vol. 9, pp. 120996–121005, 2021.

[12] U. Srilakshmi, N. Veeraiah, Y. Alotaibi, S. A. Alghamdi, O. I. Khalaf, and B. V. Subbayamma, “An improved hybrid secure multipath routing protocol for manet,” *IEEE Access*, vol. 9, pp. 163043–163053, 2021.

[13] K. Lim, K. M. Tuladhar, X. Wang, and W. Liu, “A scalable and secure key distribution scheme for group signature based authentication in vanet,” in *2017 IEEE 8th annual ubiquitous computing, electronics and mobile communication conference (UEMCON)*, pp. 478–483, IEEE, 2017.

[14] P. Vijayakumar, M. Azees, A. Kannan, and L. J. Deborah, “Dual authentication and key management techniques for secure data transmission in vehicular ad hoc networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1015–1028, 2015.

[15] S. H. Islam, M. S. Obaidat, P. Vijayakumar, E. Abdulhay, F. Li, and M. K. C. Reddy, “A robust and efficient password-based conditional privacy preserving authentication and group-key agreement protocol for vanets,” *Future Generation Computer Systems*, vol. 84, pp. 216–227, 2018.

[16] M. Bilal and S.-G. Kang, “A secure key agreement protocol for dynamic group,” *Cluster Computing*, vol. 20, pp. 2779–2792, 2017.

[17] M. Azees and P. Vijayakumar, “Cekd: Computationally efficient key distribution scheme for vehicular ad-hoc networks,” *Australian Journal of Basic and Applied Sciences*, vol. 10, no. 2, pp. 171–175, 2016.

[18] V. Kumar, R. Kumar, and S. K. Pandey, “A computationally efficient centralized group key distribution protocol for secure multicast communications based upon rsa public key cryptosystem,” *Journal of King Saud*

University-Computer and Information Sciences, vol. 32, no. 9, pp. 1081–1094, 2020.

[19] A. Mansour, K. M. Malik, A. Alkaff, and H. Kanaan, "Alms: Asymmetric lightweight centralized group key management protocol for vanets," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1663–1678, 2020.

[20] O. S. Oubbati, N. Chaib, A. Lakas, P. Lorenz, and A. Rachedi, "Uav-assisted supporting services connectivity in urban vanets," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3944–3951, 2019.

[21] H. Hamey and C. Muckenhim, "Group key management protocol (gkmp) specification," *RFC2093, Internet Engineering Task Force*, vol. 70, 1997.

[22] H. Harney and C. Muckenhirn, "Group key management protocol (gkmp) architecture," tech. rep., 1997.

[23] R. Canetti, J. Garay, G. Itkis, D. Micciancio, M. Naor, and B. Pinkas, "Multicast security: A taxonomy and some efficient constructions," in *IEEE INFOCOM'99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No. 99CH36320)*, vol. 2, pp. 708–716, IEEE, 1999.

[24] L. Mingyan, R. Poovendran, and C. Berenstein, "Optimization of key storage for secure multicast," in *35th Annual Conference on Information Sciences and Systems (CISS)*, 2001.

[25] C. K. Wong, M. Gouda, and S. S. Lam, "Secure group communications using key graphs," *IEEE/ACM transactions on networking*, vol. 8, no. 1, pp. 16–30, 2000.

[26] A. Penrig, D. Song, and D. Tygar, "Elk, a new protocol for efficient large-group key distribution," in *Proceedings 2001 IEEE Symposium on Security and Privacy. S&P 2001*, pp. 247–262, IEEE, 2000.

[27] A. T. Sherman and D. A. McGrew, "Key establishment in large dynamic groups using one-way function trees," *IEEE transactions on Software Engineering*, vol. 29, no. 5, pp. 444–458, 2003.

[28] G.-H. Chiou and W.-T. Chen, "Secure broadcasting using the secure lock," *IEEE Transactions on Software Engineering*, vol. 15, no. 8, pp. 929–934, 1989.

[29] D. Wallner, E. Harder, and R. Agee, "Key management for multicast: Issues and architectures," tech. rep., 1999.

[30] C. K. Wong, M. Gouda, and S. S. Lam, "Secure group communications using key graphs," *ACM SIGCOMM Computer Communication Review*, vol. 28, no. 4, pp. 68–79, 1998.

[31] C. K. Wong, M. Gouda, and S. S. Lam, "Secure group communications using key graphs," *IEEE/ACM transactions on networking*, vol. 8, no. 1, pp. 16–30, 2000.

[32] D. Balenson, D. McGrew, and A. Sherman, "Key management for large dynamic groups: One-way function trees and amortized initialization," tech. rep., Internet-Draft, 1999.

[33] A. T. Sherman and D. A. McGrew, "Key establishment in large dynamic groups using one-way function trees," *IEEE transactions on Software Engineering*, vol. 29, no. 5, pp. 444–458, 2003.

[34] R. Canetti, J. Garay, G. Itkis, D. Micciancio, M. Naor, and B. Pinkas, "Multicast security: A taxonomy and efficient constructions, march 1999," in *Infocom99*, vol. 13.

[35] M. Waldvogel, G. Caronni, D. Sun, N. Weiler, and B. Plattner, "The versakey framework: Versatile group key management," *IEEE Journal on selected areas in communications*, vol. 17, no. 9, pp. 1614–1631, 1999.

[36] G. R. K. Rao and G. Radhamani, *WiMAX: a wireless technology revolution*. CRC Press, 2007.

[37] S. A. Ahson and M. Ilyas, *WiMAX: Standards and security*. CRC press, 2018.

[38] S. Glisic and J.-P. Makela, "Advanced wireless networks: 4g technologies," in *2006 IEEE Ninth International Symposium on Spread Spectrum Techniques and Applications*, pp. 442–446, IEEE, 2006.

[39] B. DeCleene, L. Dondeti, S. Griffin, T. Hardjono, D. Kiwior, J. Kurose, D. Towsley, S. Vasudevan, and C. Zhang, "Secure group communications for wireless networks," in *2001 MILCOM Proceedings Communications for Network-Centric Operations: Creating the Information Force (Cat. No. 01CH37277)*, vol. 1, pp. 113–117, IEEE, 2001.

[40] T. Hardjono and B. Cain, "Secure and scalable interdomain group key management for n-to-n multicast," in *Proceedings 1998 International Conference on Parallel and Distributed Systems (Cat. No. 98TB100250)*, pp. 478–485, IEEE, 1998.

[41] A. Ballardie, "Core based trees multicast routing, protocol specification," *RFC2189*, 1997.

[42] L. R. Dondeti, S. Mukherjee, and A. Samal, "Scalable secure one-to-many group communication using dual encryption," *Computer Communications*, vol. 23, no. 17, pp. 1681–1701, 2000.

[43] S. Rafaeli and D. Hutchison, "Hydra: A decentralised group key management," in *Proceedings. Eleventh IEEE international workshops on enabling technologies: infrastructure for collaborative enterprises*, pp. 62–67, IEEE, 2002.

[44] S. Setia, S. Koussih, S. Jajodia, and E. Harder, "Kronos: A scalable group re-keying approach for secure multicast," in *Proceeding 2000 IEEE Symposium on Security and Privacy. S&P 2000*, pp. 215–228, IEEE, 2000.

[45] B. Briscoe, "Marks: Zero side effect multicast key management using arbitrarily revealed key sequences," in *International Workshop on Networked Group Communication*, pp. 301–320, Springer, 1999.

[46] I. Ingemarsson, D. Tang, and C. Wong, "A conference key distribution system," *IEEE Transactions on Information theory*, vol. 28, no. 5, pp. 714–720, 1982.

[47] M. Steiner, G. Tsudik, and M. Waidner, "Diffie-hellman key distribution extended to group communication," in *Proceedings of the 3rd ACM Conference on Computer and Communications Security*, pp. 31–37, 1996.

[48] W. Diffie and M. E. Hellman, "New directions in cryptography," in *Democratizing Cryptography: The Work of Whitfield Diffie and Martin Hellman*, pp. 365–390, 2022.

[49] K. Becker and U. Wille, "Communication complexity of group key distribution," in *Proceedings of the 5th ACM Conference on Computer and Communications Security*, pp. 1–6, 1998.

[50] Y. Kim, A. Perrig, and G. Tsudik, "Communication-efficient group key agreement," in *Trusted Information: The New Decade Challenge 16*, pp. 229–244, Springer, 2001.

[51] O. Rodeh, K. P. Birman, and D. Dolev, "Optimized rekey for group communication systems.," in *NDSS*, pp. 37–48, 2000.

[52] L. R. Dondeti, S. Mukherjee, and A. Samal, "Distributed group key management scheme for secure many-to-many communication," May 29 2001. US Patent 6,240,188.

[53] A. Fiat and M. Naor, "Broadcast encryption," in *Advances in Cryptology—CRYPTO'93: 13th Annual International Cryptology Conference Santa Barbara, California, USA August 22–26, 1993 Proceedings 13*, pp. 480–491, Springer, 1994.

[54] C. Boyd, "On key agreement and conference key agreement," in *Information Security and Privacy: Second Australasian Conference, ACISP'97 Sydney, NSW, Australia, July 7–9, 1997 Proceedings 2*, pp. 294–302, Springer, 1997.

[55] M. Burmester and Y. Desmedt, "A secure and efficient conference key distribution system," in *Advances in Cryptology—EUROCRYPT'94: Workshop on the Theory and Application of Cryptographic Techniques Perugia, Italy, May 9–12, 1994 Proceedings 13*, pp. 275–286, Springer, 1995.

[56] Y. Sun, W. Trappe, and K. R. Liu, "An efficient key management scheme for secure wireless multicast," in *2002 IEEE International Conference on Communications. Conference Proceedings. ICC 2002 (Cat. No. 02CH37333)*, vol. 2, pp. 1236–1240, IEEE, 2002.

[57] Y. Sun, W. Trappe, and K. R. Liu, "Topology-aware key management schemes for wireless multicast," in *GLOBECOM'03. IEEE Global Telecommunications Conference (IEEE Cat. No. 03CH37489)*, vol. 3, pp. 1471–1475, IEEE, 2003.

[58] L. Lin, X. Li, and Y. Cheng, "Hkm: A hybrid key management scheme for secure mobile multicast," in *2007 International Conference on Networking, Architecture, and Storage (NAS 2007)*, pp. 109–114, IEEE, 2007.

[59] Y. Wang, P. D. Le, and B. Srinivasan, "Hybrid group key management scheme for secure wireless multicast," in *6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)*, pp. 346–351, IEEE, 2007.

[60] Y. Wang, P. D. Le, and B. Srinivasan, "Efficient key management for secure wireless multicast," in *2008 Third International Conference on Convergence and Hybrid Information Technology*, vol. 2, pp. 1131–1136, IEEE, 2008.

[61] S. Gharout, A. Bouabdallah, M. Kellil, and Y. Challal, "Key management with host mobility in dynamic groups," in *Proceedings of the 3rd international conference on Security of information and networks*, pp. 186–194, 2010.

[62] S. Gharout, A. Bouabdallah, Y. Challal, and M. Achemlal, “Adaptive group key management protocol for wireless communications,” *Journal of Universal Computer Science*, vol. 18, no. 6, pp. 874–899, 2012.

[63] M. Kellil, A. Olivereau, and C. Janneteau, “Rekeying in secure mobile multicast communications,” June 21 2007. US Patent App. 10/596,786.

[64] M. L. M. Kiah and K. M. Martin, “Host mobility protocol for secure group communication in wireless mobile environments,” in *Future Generation Communication and Networking (FGCN 2007)*, vol. 1, pp. 100–107, IEEE, 2007.