The University of Bradford Institutional Repository

<http://bradscholars.brad.ac.uk>

This work is made available online in accordance with publisher policies. Please refer to the repository record for this item and our Policy Document available from the repository home page for further information.

To see the final version of this work please visit the publisher's website. Available access to the published online version may require a subscription.

Link to publisher's version: <http://dx.doi.org/10.1109/TMC.2014.2362760>

Citation: Mapoka TT, Shepherd SJ and Abd-Alhameed RA (2014) A new multiple service key management scheme for secure wireless mobile multicast. IEEE Transactions on Mobile Computing. 14(8): 1545-1559.

Copyright statement: © 2014 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other users, including reprinting/ republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted components of this work in other works. Full-text reproduced in accordance with the publisher's self-archiving policy.

A new multiple service key management scheme for secure wireless mobile multicast

T.T. Mapoka, MIEEE, S.J. Shepherd, SMIEEE, and R.A. Abd-Alhameed, SMIEEE.

Abstract— Addressing key management in mobile multicast communication is currently a booming topic due to the convergence of wireless and mobile technologies. With the proliferation of multiple group based services that are possible to co-exist within a single network, mobile subscribers could subscribe to these services concurrently while ubiquitous. However, the existing group key management (GKM) protocols intend to secure group communication for just a single group service. The GKM approaches involve inefficient use of keys and huge rekeying overheads, hence unsuitable for multiple multicast group environments. In this paper, we propose a novel GKM protocol for multiple multicast groups, called slot based multiple group key management (SMGKM) scheme. SMGKM supports the movement of single and multiple members across a homogeneous or heterogeneous wireless network while participating in multiple group services with minimized rekeying transmission overheads. Unlike conventional GKM protocols, SMGKM protocol can mitigate 1-affect-n phenomenon, single point of failure and investment pressure of signaling load caused by rekeying at the core network. Numerical analysis and simulation results of the proposed protocol show significant resource economy in terms of communication bandwidth overhead, storage overheads at the Domain Key Distributor (DKD), mobile receiver and Area Key Distributors while providing intense security.

Index Terms— Group key management, mobile multicast, security, wireless networks

1 INTRODUCTION

Multicast is a bandwidth efficient technique for delivering group-oriented applications over the internet. These include applications such as video conferencing, interactive group games, video on demand (VoD), and mobile TV services. Multicast content distribution utilizes one-to-many and many-to-many transport communication mechanism. However the development of wireless networks and emergence of portable devices like smartphones, tablets has also increased to meet the demand for these multicast applications. The evolving wireless networks such as WiMAX [1] and 3GPP [2] have standardized the multimedia broadcast/multicast service (MBMS). MBMS provide efficient delivery of broadcast and multicast services, both within a cell and within the core network. The evolved MBMS in Long Term Evolution (LTE) [3] is a handy example.

However due to open access in wireless networks, the broadcasted multicast services over the air become vulnerable to various security attacks such as eavesdropping opportunities, Denial of service (DoS), physical node capture attacks, impersonation attacks and others [4]. To deliver the multicast content securely only to the group members in wireless networks, an access control mechanism which ensures confidentiality, safeguards digital contents, and simplifies accounting for the broadcasted services is obligatory. This becomes a vital requirement

for successful commercialization of these multicast services in wireless networks. The standard way to provide access control mechanism for secure multicast communication is by using a symmetric group key, known as Traffic Encryption Key (TEK), shared only by authorised group members. Group messages encrypted with TEK can be decrypted by legitimate group members holding similar TEK therefore assuring secure group communication. Maintaining an efficient key management system is challenging due to group membership dynamics caused by member joins or leaves. This triggers update of the TEK through rekeying process. During rekeying process, the key server delivers the new TEK to the existing group members to invalidate the old TEK. This restrict access to the future (prior) messages after member (join) leaves, to satisfy forward (backward) secrecy. To efficiently reduce the key management rekeying overheads, tree GKM protocols have been extensively studied in the literature [5].

Consequently, it is predictable that in the future, multiple multicast groups will co-exist within the same network due to the emergence of various group-based applications and computationally fast mobile devices along with increased data rates for next generation wireless networks. Such a situation is probable to cause substantial key management overhead at the service provider for supporting multi-group services. Thus, the existing GKM schemes for secure wired [6] and wireless mobile [7] multicast networks will suffer from rekeying performance for cumulative multicast services because there are only targeted for a single multicast service. Fig.1 illustrates an example of a multi-service environment whereby members under different service groups subscribe to various set of multicast services offered by the service provider in an LTE network for example; Voice service and charged TV streaming with stringent and low delay sensitive to

- T.T.Mapoka is with the School of Engineering and Informatics, University of Bradford, Bradford, UK. E-mail: tmapoka@bradford.ac.uk.
- S.J.Shepherd is with the School of Engineering and Informatics, University of Bradford, Bradford, UK. E-mail: sjshepherd@bradford.ac.uk.
- R.A.Abd-Alhameed is with the School of Engineering and Informatics, University of Bradford, Bradford, UK. E-mail: raabd@bradford.ac.uk

packet delivery, web browsing and delay tolerant telemetry services like emails based on best effort delivery. Suppose each multicast service is independently controlled by a single GKM protocol according to the existing GKM schemes. If a member participating in multi-services dynamically leaves or joins all the subscribed services, all the affected services would require independent rekeying procedure hence triggering significant rekeying overhead. In addition to host mobility in mobile environments, rekeying overhead is induced twofold since a handover member is considered leaving the currently serving subnet followed by join at the target subnet while maintaining similar services subscribed. Therefore some lightweight access control mechanism need to be addressed to prevent service latency while reducing rekeying latency concurrently.

To solve the rekeying complexity as multicast services cumulate in a single network, we propose a novel slot based multiple group key management (SMGKM) protocol for managing both single-move and multi-moves across a wireless network while seamlessly participating in multi-services with minimized rekeying transmission overheads, reduced communication and storage overheads, no single point of failures, no 1-affect-n phenomenon and optimized signalling load at the core network. In SMGKM the key management tasks is offloaded to the intermediate cluster managers called Area Key Distributors which establish the necessary key management keys. SMGKM integrate our concept of session key distribution list(SKDL) introduced in [8] for fast and secure authenticated handover along with initial key establishment. SMGKM employ a lighter symmetric encryption suitable for resource constraint mobile devices than heavier asymmetric effort. Compared to the existing schemes, SMGKM save enormous communication bandwidth utilization in the presence of multi-handoffs in multi-services.

Fig.1. Multiple multicast service groups' illustration

In what follows is related work and reference framework which is discussed in section 2 and 3 respectively. Section 4 describes the SMGKM novel rekeying strategy in detail on member handoff with its analytical model in section 5. The performance analysis of the SMGKM in terms of rekeying transmission, communication and storage overheads is described in section 6. Finally section 7

concludes the paper.

2 RELATED WORK

Traditional GKM protocols addressing rekeying over wired networks are divided in to centralized, decentralized and contributory schemes [6]. Centralized schemes rely on the centralized server known as the Domain Key Distributor (DKD) for generation and distribution of encryption keys. Contributory scheme has no explicit DKD, thus group members collaborate for group key establishment. Decentralized schemes partition the group in to subgroups each managed by subgroup managers in order to equally distribute the key management tasks hence scalability. Work in [9] further categorizes the GKM as common TEK and Independent TEK per subgroup approaches depending on how the TEK is distributed in the framework. Common TEK approaches such as in [5, 10, 11] utilize one TEK for all group members and commonly suffer from 1-affect-n phenomenon; thus rekeying of the new TEK affect all the members subscribed to the same group in the entire network whenever a membership change occurs. Independent TEK per subgroup approaches try to alleviate the 1-affect-n phenomenon caused by common TEK approaches such as in [12], by enabling each subgroup to independently manage its own TEK, thus rekeying of the new TEK is localized within the affected subgroup during membership change. However the GKM protocols did not consider host mobility during their implementation though they cannot be extended to wireless mobile environment directly.

In order to address rekeying in wireless mobile environment, few GKM protocols [13-16] have been proposed recently. In addition to dynamic membership change considered for GKM protocols in wired networks, these protocols consider dynamic location change of members over a widely distributed wireless network while seamlessly receiving subscribed multicast services securely. The protocols adopt a decentralized framework for scalability. Work in [7] also categorized them according to common TEK [13-15, 17] and Independent TEK per subgroup [16] approaches as described in [9] to address similar rekeying issues. However, none of the GKM schemes previously proposed address rekeying for multiple group services. In [18] various rekeying strategies proposed only considers a single multicast service. If for instance M_1 in Fig.1 perform handoff between clusters i and v while maintaining active subscribed services, voice and sports, the rekeying process is triggered independently for the affected three service groups 1, 2 and 3 in both the old and the target cluster when Baseline rekeying strategy (BR) is used. Though forward and backward secrecy is guaranteed in BR, it leads to extensive rekeying overhead with long service disruptions. Immediate rekeying (IR) strategy solves this problem by rekeying only the local area keys, however it gives huge rekeying overhead whenever members repeatedly handover. A type of Delayed rekeying strategy named (First Entry Delayed Rekeying + Periodic (FEDRP)) alleviate IR rekeying problem by introducing mobility lists to track and manage host mobility. Hence-

forth, rekeying is only performed at the target cluster for backward secrecy since a handover member is recorded in the previous cluster list as still valid to the session. When a member finally leaves, all the clusters previously visited by the leaving member undergo rekeying hence causing considerable rekeying overhead for multiple services case. Most GKM schemes such as Declene et al [13], GKMF [14, 17], Kellil et al [15] adopt DR strategy for efficient rekeying though inefficient for multi-services co-existing in a single network. The schemes also suffer from 1-affect-n phenomenon which requires all clusters commit to the updated TEK during rekeying. For this reason, we propose a novel key management rekeying strategy, not realized in the design of conventional approaches to address security for multi-service groups subscribed by multi-users as illustrated in Fig.1. However, this paper only considers dynamic member location change of mobile hosts subscribed to multiple subscriptions without considering dynamic membership change which is also applicable.

3 REFERENCE FRAMEWORK

Our framework adopts a two tier decentralized framework similar to [13-15] as shown in Fig.2.

Fig.2. Reference framework

The first level is the domain level which is the core wired part consisting of Domain Key Distributor for initial key management and authentication procedures. The second level is the area level which is the wireless part consisting of multiple clusters each of which are managed by the Area key Distributor (AKD) independently. Each cluster contains a set of members subscribed to diverse multicast services who dynamically perform handoff across widely distributed clusters under homogeneous or heterogeneous vendors. However, the framework adopts independent TEK per cluster to alleviate 1-affect-n phenomenon and to localize rekeying process. If rekeying process of the TEK is triggered due to joins or leaves emanating from mobility, this is handled locally without disturbing the entire system. In order to unburden the key management and authentication phases of the SMGKM from centralized DKD, we allow the AKDs to verify moving sub-

scribers, generate, update and distribute the service encryption keys. We also introduce AKD to AKD link in the framework similar to X2 interface in LTE [3] to improve the handoff rekeying performance by reducing the investment pressure at the core of the network while giving DKD scalability. Similar to DR, each AKD maintains secure mobility list called Session Key Distribution list (SKDL) to track mobility and reduce the need for rekeying when members handoff while maintaining the subscribed services.

This generic framework can be mapped in to all IP next generation network architectures like LTE and WiMAX. In LTE, the Mobility Management Entity (MME) can perform the role of the DKD and in WiMAX the Connectivity Service Network (CSN) AAA server can perform the DKD duties while the Access Service Network (ASN) gateway can act as the SP. The eNodeB (eNB) or BS correspond to the AKDi. Though the mappings can be unique, key management, mobility management and authentication phases are maintained in SMGKM.

3.1 Initial key distribution

Using our SKDL concept introduced in [8], DKD initially derives the necessary cryptographic keys on group setup and the rest is handled at the cluster level. After successfully registration of mobile receivers M_i subscribed to diverse multicast services and knowing their mobility pattern, DKD initially derives the M_i short term individual AKD_i specific session keys ($SK_{M_i_AKD_i}$) depending on the M_i mobility pattern. DKD also derives the M_i unique long term authentication key (AK_{M_i}) which is embedded on the mobile device smartcard. The DKD then generate secure session key distribution list ($SKDL_i$) for the particular AKD_i with rows corresponding to the number of registered M_i under cluster area i . Table 1 shows an illustrate example of the generated $SKDL_i$ where priority numbers determines either a high speed M_i subscribed to services with stringent delay and sensitive to packet delivery or low speed M_i subscribed to delay tolerant services.

TABLE 1
SESSION DISTRIBUTION LIST KEY

#	AKD_ID AKD_i	ES $_{SAi}[SK_{M_i_AKD_i}]$	MAC $_{SAi}$
1	ID AKD_1	ES $_{SA1}[SK_{M_1_AKD_1}]$	MAC $_{SA1}[ID_{AKD_1} \parallel ES_{SA1}[SK_{M_1_AKD_1}]]$
2	ID AKD_2	ES $_{SA2}[SK_{M_2_AKD_2}]$	MAC $_{SA2}[ID_{AKD_2} \parallel ES_{SA2}[SK_{M_2_AKD_2}]]$
3	ID AKD_3	ES $_{SA3}[SK_{M_3_AKD_3}]$	MAC $_{SA3}[ID_{AKD_3} \parallel ES_{SA3}[SK_{M_3_AKD_3}]]$
:	:	:	:
n	ID AKD_n	ES $_{SAi}[SK_{M_i_AKD_i}]$	MAC $_{SAi}[ID_{AKD_i} \parallel ES_{SAi}[SK_{M_i_AKD_i}]]$

Each $SKDL_i$ row is encrypted using pairwise security association key SA_i shared between the AKD_i and the DKD for securely pushing the corresponding $SKDL_i$ rows to the AKD_i at cluster i where M_i currently resides. The row information is also integrity protected using unique MAC to alleviate replay attacks. Thus each AKD_i has the capability to modify its own rows without affecting the rows for its neighbors. If members newly join the network, the DKD becomes the initial step to derive its $SK_{M_i_AKD_i}$ then forward the M_i particular row to its current location so that the corresponding AKD_i update its $SKDL_i$ rows. Each AKD_i on receiving the $SKDL_i$ rows, it can securely establish N TEK $_{ij}$ shares (TEK_{ij}) for N services using Key derivation function (KDF) such as SHA1 [20] without

involvement of the DKD hence giving DKD scalability.

3.2 System assumptions and notations

The design of the SMGKM protocol is based on the following assumptions specified in [14] and the notations used are described in Table 2:

- The DKD, AAA server and the Service provider (SP) are collocated for simplicity.
- The keys are already established as shown in Fig.2 at initial group setup and the DKD, SP and the AKDs are aware of the Key update slot (KUS) generation format and the maximum number of multicast services provided by the SP.
- Registration phase of the M_i is performed successfully and the registered M_i already hold the specific Authentication keys (AK_{Mi}).
- All key distributors (DKD and AKDs) are trustworthy and reliable and all the MNs trust them.
- All the AKDs can generate new KUS from the initial KUS quickly depending on the number of services affected when mobility start to occur.
- The AKDs are mobility aware and can only store the necessary SKDL_i parameters for the M_i they are currently serving otherwise delete the unnecessary. Each SKDL_i row is integrity protected using the MAC.
- The AKD and M_i have the capability of deriving the necessary keys using backward key separation technique without involving the DKD.

TABLE 2
NOTATIONS, SIGNIFICANCE AND FUNCTION

Notation	Significance	Function
AKD _i	Area Key Distributor of cluster i	Grant access to M_i , TEK _i derivation, TEK _{i,j} mapping in to the KUS and TEK _{i,j} updating during rekeying process
DKD	Domain Key Distributor	Initial KUS and SKDL generation and updating the SKDL
M_i	Mobile Node in cluster i	Existing moving member in cluster i
M_v	Mobile node in the visited cluster v	Existing moving member in cluster v
$[M]_k$	Message M encrypted with symmetric shared key K	Symmetric Encryption of message M
KUS _i	Key Update Slot in cluster i	Determines number of affected sessions which require TEK update with the corresponding number of M_i per session.
SKDL _i	Session Key Distributor Lists on cluster i	Maintain a list M_i with their already derived SKMi by the DKD at the corresponding AKD _i
SK _{Mi}	M_i specific session key	M_i specific shared pairwise symmetric key for encrypting messages between the AKD _i and the M_i
TEK _{i,j}	Traffic Encryption Key of cluster i in the jth multicast group session Group Key in cluster i	Specific symmetric key for encrypting and decrypting messages between the AKD _i and the M_i
TEK _i	Group Key in cluster i	Group delay derived by the AKD _i containing multiple TEK _{i,j} shares for various group services
AK _{MNi}	Authentication Key	Authentication Key held by the M_i used to derive the SKi knowing the ID _{AKDi}
ID _{AKDi}	Identity of the AKD _i	Used by the M_i to derive the SK _{Mi}
ID _{Gj}	Identity of the multicast group session j	Used by the AKD _i to identify the particular service M_i is subscribed to.
$\Rightarrow$	Multicast/broadcast message	Used during rekeying to deliver the new TEK _{i,j} to all M_i holding old TEK _{i,j} share with the single transmission.
$\rightarrow$	Unicast message	Used during rekeying to deliver the new TEK _{i,j} to all M_i holding a valid SKMi individually

3.3 System security requirements

In multicast applications with multiple services, members not only dynamically join or leave the services as addressed in single service scenario. However, members can handoff between homogeneous or heterogeneous wireless network clusters at various speeds. This additionally complicates rekeying more than in single membership scenario. A novel rekeying strategy addressing single and multi-moves in multi-services is presented in section 4. However, let's first denote $S = \{s_1, s_2, s_3, \dots\}$ as a set of multicast services provided by the SP. From the access control viewpoint, multicast Service Groups (SG) is a set of members M_i accessing exactly same set of multicast services. We denote S as s_N , where N equals total number of multicast services and denote MGs as $\{G_1, G_2, \dots, G_K\}$, K denotes number of SGs. Therefore it can be noticed from Fig.1 that S have overlapping membership compared to SGs since members may subscribe to many services. This proves that $K \leq 2^N - 1$. Using the accessibility matrix $A_{i,N}$, we can easily identify the access relationship between the multicast services and the MGs. $A_{i,N}$ denote the element on the i^{th} row and N^{th} column of the accessibility matrix A . From Fig.1, members ($M_1, \dots, M_6$) can access any of the four services {voice (s_1), mobile TV (s_2), stock quote (s_3), email (s_4)} such that:

$$A_{i,N} = \begin{cases} 1, & \text{if } M_i \text{ access service } s_N \\ 0, & \text{otherwise} \end{cases}$$

, where i is the rows representing members in G_K and N represent columns of subscribed services. Based on the accessibility matrix $A_{i,N}$, let's assign the G_K to access set of services s_N such that members in G_K access service set $\{s_N\}$. For example, in Fig 1, G_1 access $\{s_1, s_2\}$, G_2 access $\{s_2, s_3\}$, G_3 access $\{s_2, s_3, s_4\}$, G_4 access $\{s_3, s_4\}$, G_5 access $\{s_4, s_5, s_6\}$ and G_6 access $\{s_4, s_6\}$ for $K=1 \dots 6$. Clearly, due to overlapping memberships $\exists M_i \in G_K: A_{i,N} = 1$. Thus members may subscribe to more than one service according to their access privilege defined by accessibility matrix A . Thus, based on the definition above, the group size of each service s_N should satisfy:

$$n(s_N) = \sum_{j=0}^K n(G_j) \times A_{i,N} \quad (1)$$

where $n(s_N)$ denote the total number of members subscribed to service N and $n(G_j)$ is the number of members under a specific SG. Members in G_K , only have access to service $\{s_N, \forall_N: A_{i,N} = 1\}$, so when M_i handoff from cluster i to the target cluster v , rekeying should be performed to satisfy the following key management security requirements such that

- M_i is restricted from accessing the future content of the subscribed services $\{s_N, \forall_N: A_{i,N} = 1\}$ and $\{s_N, \forall_N: A_{j,N} = 0\}$ to satisfy forward secrecy requirement in the previous cluster i .
- M_i is restricted from accessing the prior content of the subscribed services $\{s_N, \forall_N: A_{i,N} = 0\}$ and $\{s_N, \forall_N: A_{j,N} = 1\}$ to satisfy backward secrecy requirement in the target cluster v .

4 NOVEL REKEYING STRATEGY

To simplify the access privilege for each service, Key Update slot (KUS) for N multicast services is initially generated by the trusted DKD on group setup depending on the number of members under G_K . The KUS format is partitioned in to slots of length l bits equivalent to N various services as shown in Fig. 3.

Each slot l determines the total number of members from a combination of M_{G_k} participating in service N where $A_{i,N} = 1$.

Fig. 3 Key update slot format

Moreover, each slot can dynamically increment or decrement by 1 whenever a join or leave occur respectively or reset to zero if no member is subscribed to service N . For example, using Fig.1 with 12 members, $\{M_1, M_2\} \in G_1$ access $\{s_1\}$, $\{M_1, M_2, M_3, M_4, M_5, M_6\} \in G_1, G_2, G_3$ access $\{s_2\}$, $\{M_3, M_4, M_5, M_6, M_7, M_8\} \in G_2, G_3, G_4$ access $\{s_3\}$, $\{M_5, M_6, M_7, M_8, M_9, M_{10}\} \in G_3, G_4, G_5$ access $\{s_4\}$, $\{M_9, M_{10}, M_{11}, M_{12}\} \in G_5, G_6$ access $\{s_5\}$ and $\{M_9, M_{10}, M_{11}, M_{12}\} \in G_5, G_6$ access $\{s_6\}$. Therefore the initial KUS generated acquiescent to Fig.1 for the corresponding $n(s_N)$ is shown in Fig.4 with $\{2, 6, 6, 6, 4, 4\}$ members in the service slots $\{s_1, s_2, s_3, s_4, s_5, s_6\}$ respectively.

Fig.4. Initial generated KUS for all members in all clusters

For each slot set to be l bits long, the maximum allowable number of members per service slot can be evaluated as 2^l which gives scalability. Thus from equation (1), $n(s_N) \leq 2^l$. Thus the total size in bits of the generated KUS can be evaluated as $N \cdot l$. The assumption is that, all the AKDs for the corresponding clusters are already aware of the KUS format and the DKD have securely distributed the initial KUS to the AKDs as shown in Fig.55 and Fig. 66 respectively depending on the number of members M_i residing under them.

Fig.5. KUS update at cluster i

Fig. 6. KUS update at cluster v

4.1 Key generation and data confidentiality

Data confidentiality for multiple data streams in the same network can be applied to the generated KUS. Thus in SMGKM, the AKD_i initially derive service specific group key (TEK_i) which is split in to multiple TEK shares $\{TEK_{i,1}, TEK_{i,2}, \dots, TEK_{i,j}\}$ corresponding to N services such that $TEK_{i,j} \ll TEK_i$, where i denotes the cluster and j denotes a particular service. These shares are then mapped in to the KUS format (TEK mapping) as illustrated in

Fig.77.

Fig.7. TEK KUS mapping and data confidentiality

In this case each service is encrypted using separate shares denoted as $TEK_{i,j}$ which is common to all $M_i \in G_K$ subscribed to service j . Therefore the duty of key management is to securely update and re-distribute $\{TEK_{i,j} \forall N: A_{i,N} = 1\}$ to $M_i \in G_K$ during rekeying since members may subscribe to multiple services concurrently regardless of location. As shown in Fig.2, members belonging to unique clusters hold cryptographically separate TEK shares specific to the cluster so that key compromise is limited to the affected cluster.

4.2 Secure data transmission

Assuming the SP hold valid $TEK_{i,j}$ shares, it securely transmit the services subscribed to the corresponding M_i under each cluster using a specific bandwidth for that service. Let $BW(s_N)$ denote the bandwidth used to transmit service s_N , let $C(n(G_K))$ be the cost of transmitting a unit data to n members M_i for $i=1, \dots, n$ in G_K via multicast. We study the two methods that can be used to efficiently transmit the services illustrated in Fig 1 to the corresponding M_i in G_K . In *method 1*, the total communication cost ($TC_{METHOD1}$) of transmitting the multicast services can be evaluated as:

$$\begin{aligned} TC_{METHOD1} &= BW(s_N) \cdot C\left(\sum_{j=1}^K n(G_j)\right) \\ &= BW(s_1) \cdot C(n(G_1)) \\ &\quad + BW(s_2) \cdot C(n(G_1) + n(G_2) + n(G_3)) \\ &\quad + BW(s_3) \cdot C(n(G_2) + n(G_3) + n(G_4)) \\ &\quad + BW(s_4) \cdot C(n(G_3) + n(G_4) + n(G_5)) \\ &\quad + BW(s_5) \cdot C(n(G_5) + n(G_6)) \\ &\quad + BW(s_6) \cdot C(n(G_5) + n(G_6)). \end{aligned}$$

In *method 2*, the total communication cost ($TC_{METHOD2}$) of

transmitting the multicast services can be evaluated for each service as:

$$\begin{aligned}
T_{METHOD2} &= C(n(G_K)) \cdot \sum_{j=1}^{N-1} BW(s_j) \\
&= (BW(s_1) + BW(s_2)) \cdot C(n(G_1)) \\
&+ (BW(s_2) + BW(s_3)) \cdot C(n(G_2)) \\
&+ (BW(s_2) + BW(s_3) + BW(s_4)) \cdot C(n(G_3)) \\
&+ (BW(s_3) + BW(s_4)) \cdot C(n(G_4)) \\
&+ (BW(s_4) + BW(s_5) + BW(s_6)) \cdot C(n(G_5))
\end{aligned}$$

If we compare $TC_{METHOD1}$ and $TC_{METHOD2}$, it can clearly be observed that $TC_{METHOD2} < TC_{METHOD1}$ if the fact $C(x + y) < C(x) + C(y)$ is applied.

Therefore SMGKM can transmit multiple services with optimised bandwidth utilization without sacrificing security using $TC_{METHOD2}$ with aggregated service bandwidth. However the conventional approaches use $TC_{METHOD1}$ which require huge bandwidth for secure data transmission. However $TC_{METHOD1}$ can be optimized by applying tree based approach [5].

4.3 Member handoff with backward secrecy

To address the rekeying problem, only member handoff with backward secrecy is considered in our system since members maintain session continuity while changing point of attachment to the network. This makes forward secrecy requirement unnecessary at the old cluster i . Therefore the joining M_i at the target cluster v should be restricted from accessing the prior content of the subscribed services $\{s_N, \forall_N: A_{i,N} = 0\}$ and $\{s_N, \forall_N: A_{j,N} = 1\}$ to satisfy backward secrecy requirement. By using the illustrative example in Fig.2, whenever M_1 discover low signal strength (P_i) from AKD_i of cluster i , and high signal strength (P_v) from AKD_v of cluster v ($P_i < P_v$).

Step 1: M_1 simultaneously send a *move_notify* message to both AKD_i and the target AKD_v encrypted under its derived $SK_{ML_AKD_i}$ to AKD_i , i.e.

$$M_1 \rightarrow AKD_i: \{move_notify\}_{SK_{ML_AKD_i}}$$

Step 2: AKD_i on receiving the *move_notify* message it verifies $SK_{ML_AKD_i}$ against the already derived $SK_{ML_AKD_i}$ by the DKD stored in the $SKDL_i$ row. If M_1 derived $SK_{ML_AKD_i}$ matches, then the AKD_i can decrypt the *move_notify* message. However AKD_i wait for a period T_{up} to accommodate more incoming notification requests.

Step 3: AKD_i verify the number and ID_{G_j} of services subscribed to M_1 along with the target AKD_v from the notification message. Note that AKD_i cannot modify the $SKDL_v$ rows of the target AKD_v since the assumption is that each row is integrity protected using AKD_i specific MAC, i.e. each AKD can access and modify its rows. This is also beneficial for secure accounting purposes not considered in this paper.

Step 4: AKD_i notices that M_1 is subscribed to s_1 and s_2 from Fig.1 and start generating the new KUS as shown in Fig.5 from the initially generated KUS in Fig.44 after T_{up} elapse. The affected services s_1 and s_2 require backward secrecy rekeying of $\{TEK_{i,j} \forall_N: A_{i,N} = 1\}$ to all $M_i \in G_1, G_2, G_3$ at the target cluster after the KUS update process in Fig. 6 by the target AKD_v .

Step 5: AKD_i securely transmit the new KUS to the

target AKD_v along with the M_1 row in the $SKDL_v$ via the fast AKD_i - AKD_v link using context transfer protocol (CXTF) [19] without involving the DKD. This provides better support for accelerating handoff.

Step 6: AKD_i finally wait until M_1 completely handoff then revokes the M_1 row from its $SKDL_i$ for security purposes. Note that each AKD_i only stores information of the existing members M_i they are currently serving to enhance storage complexity as described in section 6.3.

Step 7: The target AKD_v on receiving the new KUS along with its rows, it also performs integrity checks of its rows received before obtaining the already M_1 derived $SK_{ML_AKD_v}$ specific to the target cluster v . From the received new KUS, the AKD_v also determines the services subscribed by M_1 then undergo KUS update process in Fig. 6. For the affected services requiring $TEK_{i,j}$ shares update, AKD_v update the key shares and wait until M_1 signal its interest to move in to cluster v .

Step 8: On arrival at the crossover region towards the target cluster v where ($P_i < P_v$), M_1 knowing the AK_{M_i} and detecting the ID_{AKD_v} , it derives the $SK_{ML_AKD_v}$ using one of prominent KDFs such as SHA1 [20]. Note that the TEK shares used by M_1 for the same services at the old cluster i are revoked after total handoff to reserve memory for the new TEK shares for the same services from AKD_v . Thus M_1 then send a *move_notify* message encrypted under the newly derived $SK_{ML_AKD_v}$ shared with the AKD_v .

Step 9: The AKD_v undergo $SK_{ML_AKD_v}$ verification process described in step 2. After verifying the M_1 request as well as the services subscribed to M_1 , it securely transmit the new $TEK_{i,v}$ shares for the affected services (s_1, s_2) according to the received KUS from AKD_i via unicast response message to M_1 encrypted under $SK_{ML_AKD_v}$ and via multicast response message to the existing members M_v subscribed to the same set of affected services (s_1, s_2) encrypted with the *old* $TEK_{i,v}$ shares. i.e.

- 1) $AKD_v \rightarrow M_1: \{move_response(new_TEK_{i,v})\}_{SK_{ML_AKD_v}}$
- 2) $AKD_v \rightarrow M_v: \{move_response(new_TEK_{i,v})\}_{old_TEK_{i,v}}$

This therefore guarantees backward secrecy requirement of the SMGKM protocol. The assumption is that the AKD_v notifies the DKD collocated with the SP of each rekeying so that the SP utilize the updated TEK shares for secure data transmission if no join or leave event occur within the T_{up} period. The overall signal flow for the SMGKM novel rekeying strategy is summarized Fig.8.

Fig.8. Signal flow for our SMGKM protocol

4.4 SMGKM rekeying comparison with other protocols

This section compares the SMGKM rekeying signalling flow against that of conventional protocols targeted for a single service. We describe to the conventional protocols, BR, IR, GKMF and Kellil et al described in section 2 as key-request protocols. Key-request protocols require handover synchronized signaling with the DKD during handoff. Thus key derivation of the service group keys during rekeying process involves the centralised DKD which is single point of failure. This makes the key-request schemes potentially slower than SMGKM because the signalling does not traverse a long path to the DKD which could be far from AKDs. Since the key-request schemes are targeted for a single service, the rekeying signalling load at the core network becomes considerable especially if multi-services co-exist in the same network hence lacking DKD scalability compared to SMGKM. SMGKM requires more memory at the AKDs than key-request schemes since the key management and authentication phases are offloaded to the AKD level. Therefore we can infer that, key-request schemes are AKD memory optimized with high rekeying signalling load between the AKD-DKD links whilst SMGKM has optimized rekeying signalling load between the AKD-DKD links with high memory requirement at the AKD.

The AKD-AKD communication link is introduced in SMGKM for fast handoffs with minimal service disruptions and minimized rekeying delays. Thus the load on the DKD is independent of the AKD-AKD handover frequency and also partly independent of number of AKDs in the network. With cryptographically separate keys at each cluster, rekeying is localized hence mitigating the 1-affect-n phenomenon in key-request schemes. Furthermore SMGKM introduces the use of secured list called the SKDL compared to GKMF, KELLIL and FEDRP to prevent rekeying at the old cluster i on member leave and handoff. Key request schemes can be considered the special method of the SMGKM, where the SKDL includes only one SK_{M_i, AKD_i} at a time. It would mean that either the old AKD_i or target AKD_v request SKDL row key update from the DKD hence resembling key request schemes. The generic comparisons and number of rekeying signalling messages flow for each protocol is summarised in Table 3 and Table 4 respectively.

TABLE 3
SMGKM COMPARISON WITH KEY REQUEST PROTOCOLS

Evaluation Criteria	DeCleene et al			GKM	KEL-LIL	SMGKM
	BS	IR	FEDRP	F		
Decentralized framework	Y	Y	Y	Y	Y	Y
Number of layers	2	2	2	2	2	2
Forward secrecy on member move	Y	Y	N	N	N	N
Backward secrecy on member move	Y	Y	Y	Y	Y	Y
Rekeying of multiple visited areas after leave	Y	Y	Y	Y	Y	N
1-affect-n phenomenon	Y	Y	Y	Y	Y	N
Localize rekeying at member move	N	Y	Y	Y	Y	Y
AKD to AKD communication link	N	N	N	N	N	Y
Support multiple group services	N	N	N	N	N	Y
Single point of failure	Y	Y	Y	Y	Y	N
DKD Scalability	N	N	N	N	N	Y
Use of list to manage mobility	N	N	N	Y	Y	Y
Authentication at move	N	N	N	Y	N	Y

TABLE 4
COMPARISON OF REKEYING SIGNALING LOAD ON HANDOFF

FRAME-WORK	AKD _i ↓ DKD	AKD _v ↓ DKD	M _i ↓ ADK _i	M _i ↓ ADK _v	AKD _i ↓ ADK _v	M _v ↓ ADK _v	M _{ii} ↓ ADK _i
BS	3	3	1	3	0	3	3
IR	1	2	1	2	0	1	1
FEDRP	1	2	1	2	0	1	0
GKMF	2	2	1	2	0	1	0
KELLIL et al	1	1	1	2	0	1	0
SMGKM	0	1	1	2	1	1	0

5 SMGKM ANALYTICAL MODEL

In this section we formulate the analytic model of the fundamental system consisting of two subgroup clusters i and v , where v is the target cluster of cluster i as shown in Fig.2. Individual clusters are managed independently by the corresponding AKD_i . The analytical model is used to estimate the rekeying overhead induced when both the clusters are merged together due to member handoff from cluster i managed by AKD_i to cluster v managed by AKD_v while maintaining s -multiple multicast services. Thus members are considered as leaving cluster i followed by join at cluster v . Lets evaluate the rekeying overhead induced by the leave operation due n members leaving cluster i and the rekeying overhead induced by the join operation due n members re-joining cluster v on handoff while seamlessly maintaining session continuity. We denote the total rekeying messages overhead as RO_T .

Lets L_{ki} and J_{kv} denote the rekeying messages induced by leave operation in cluster i and join operation at cluster v respectively at steady state probability k . The expected values of L_{ki} and J_{kv} can be denoted as $E[L_{ki}]$ and $E[J_{kv}]$ respectively.

Almeroth et al [21, 22] proved that dynamism of certain multicast services over the MBone can be modeled. From the statistical viewpoint, members arriving in multicast services follow Poisson process with rate λ (arrivals/time unit). Assuming that the service time for AKDs are independent, the membership duration of a member in the group service follows an exponential distribution with a mean duration $1/\mu$ time units. The mean number

of concurrent active members under a specific cluster can therefore be evaluated as λ/μ .

Let's now apply this model to the SMGKM framework with independent TEK per cluster. Similar to [23] the assumption is that individual cluster AKDs manages parameters λ and μ independently and the clusters are probable to be joined by moving members. Furthermore, we assume that the parameters λ and μ adjust over time due to host mobility hence making the AKDs to alter its estimations of λ and μ every θ time units for estimating the rekeying overhead. We then model our system using Markov process [23]:

- If $N = \{0, 1, 2, 3, \dots\}$ denote the system state analogous to the number of concurrent active members in a specific cluster,
- If π_k denote the steady state probability that $N = k$, where π_k is evaluated from [24] as:

$$\pi_k = \frac{\rho^k}{k!} \cdot e^{-\rho} \quad (2)$$

$$\text{where } \rho = \frac{\lambda}{\mu}.$$

Now the expected number of rekey messages $E[R_{\lambda, \mu}]$ per unit time can be denoted as:

$$E[R_{\lambda, \mu}] = \lambda \sum_{k=0}^{\infty} \pi_k (E[L_{ki}] + E[J_{kv}]) \quad (3)$$

To simplify the above expression in (3), we need to approximate π_k as proved in [23] that approximating π_k as a δ -function at its mean gives the best approximation of

$$\pi_k \approx \delta(k - \frac{\lambda}{\mu}) \quad (4)$$

$$\text{where } \delta(k - \frac{\lambda}{\mu}) = \begin{cases} 1, & \text{if } k = \frac{\lambda}{\mu} \\ 0, & \text{otherwise} \end{cases}$$

Therefore $E[R_{\lambda, \mu}]$ can now be presented as:

$$E[R_{\lambda, \mu}] = \lambda \sum_{k=0}^{\infty} \delta(k - \frac{\lambda}{\mu}) (E[L_{ki}] + E[J_{kv}])$$

This reduces to:

$$E[R_{\lambda, \mu}] = \lambda (E[L_{\rho_i}] + E[J_{\rho_v}]) \quad (5)$$

where $\rho_i = \lambda_i / \mu_i$ and $\rho_v = \lambda_v / \mu_v$.

Since SMGKM adopts cryptographically separate keys per cluster rekeying process is triggered in the specific cluster where membership change occurs, thus the total rekeying overhead RO_T can be approximated by

$$RO_T = E[R_{\lambda_i, \mu_i}] + E[R_{\lambda_v, \mu_v}] \quad (6)$$

Note that merging the clusters i and v due to member mobility causes mutual impact overhead which we denote as $E[M_{i,v}]$. $E[M_{i,v}]$ defines that membership change that occurs in cluster i has an impact on the target cluster v (merged with i), and vice-versa. The first case is when member M_i leaves cluster i at rate λ_i followed by re-join at cluster v at rate λ_v as shown in Fig.2. Thus $E[M_{i,v}]$ is caused by $E[IL_{ki}]$ and $E[IJ_{kv}]$ which are the impact rekeying costs at cluster i and v respectively at steady state k . Following the steady state property of Markov process as

in (2) and (5), we obtain $E[M_{i,v}]$ as:

$$E[M_{i,v}] = \lambda_i E[IJ_{\rho_v}] + \lambda_v E[IL_{\rho_i}] \quad (7)$$

5.1 Applying the analytical model to the actual rekeying approach

Now we apply the two prominent categories of rekeying approaches proposed in literature to the analytical model. These are pairwise rekeying [25] and Logical Key hierarchy (LKH) [5] rekeying approaches.

In *pairwise rekeying approach*, all the members subscribed to similar group services share common $TEK_{i,v}$ shares. Whenever a member re-joins at the target cluster v , the AKD_v multicast the new $TEK_{i,v}$ shares for the affected services (encrypted with the old $TEK_{i,v}$) to the existing members M_v under cluster v and unicast the new $TEK_{i,v}$ shares to the new joining member M_i from cluster i encrypted by M_i derived secret key (SK_{M_i, AKD_v}). Thus the rekeying overhead at the target cluster v when M_i joins gives two messages to ensure backward secrecy, i.e.

$$E[J_{\rho_v}] = 2 \quad (8)$$

However for n members moving while participating in s services, the rekeying overhead at cluster v gives:

$$E[J_{\rho_v}] = O(n) + s \quad (9)$$

where $n = \rho_v$.

To guarantee forward secrecy at the previous cluster i , whenever member M_i leaves, the old $TEK_{i,j}$ is considered compromised and hence the AKD_i determine the affected services and distribute their new $TEK_{i,j}$ shares to remained members in cluster i (M_{ii}) encrypted under the individually derived SK_{M_i, AKD_i} that it shares with it. Thus the number of rekeying messages on leave is equivalent to the group size i.e.

$$E[L_{\rho_i}] = \rho_i \quad (10)$$

In *LKH rekeying approach*, a full balanced key graph tree of degree d is used to approximate the rekeying overhead at any time. For join operation at the target cluster v , the AKD_v where the join occurs should renew each key that is on the path from the leaf that represents the secret key/individual key associated to the new joining member to the root of the tree which denotes the TEK. Consequently, each key from the leaf of the new member to the root is transmitted twice, i.e. transmitted to the new joining member through unicast encrypted under the child key known to the new member, and via multicast to the members that share the node encrypted under its old version. Thus, the rekeying messages using the LKH with branching factor d under cluster v gives:

$$E[J_{\rho_v}] = 2 \log_d(\rho_v) \quad (11)$$

Therefore for n members moving while participating in s services, we obtain the reduced rekeying messages from equation (9) as:

$$E[J_{\rho_v}] = \log_d(\rho_v) + s \quad (12)$$

For leave operation scenario in cluster i where a hand-

over member is considered leaving, the keys on the path from the leaf that represents the leaving member to the root are considered compromised hence requiring update each of them. Each key from the leaf of the leaving member is transmitted through multicast encrypted under its d child keys to the remaining members sharing its d child keys. We now obtain the reduced rekeying messages from (10) using the LKH of degree d at cluster i as:

$$E[L_{\rho_i}] = d \log(\rho_i) \quad (13)$$

Now considering $E[M_{i,v}]$ in (7), when member(s) subscribed to s services joins cluster v on handoff, the target AKD_v distribute the new TEK shares for the affected services via multicast (one message per service) to its members (M_v) encrypted with the old TEK shares which are not compromised. Hence,

$$E[I_{\rho_v}] = s \quad (14)$$

However the computation of $E[L_{\rho_i}]$ relies of the actual rekeying approach. If *pairwise rekeying approach* is used on member leave due to handoff from cluster i , the old TEK shares for the affected services are considered compromised and hence the AKD_i sends the new TEK shares encrypted under individual secret key that it shares with its $\lambda_i / \mu_i(\rho_i)$ members.i.e.

$$E[I_{\rho_i}] = \rho_i \quad (15)$$

If *LKH rekeying approach* is used on member leave due to handoff from cluster i , we approximate the full key tree with branching factor d at any time. In this case the old TEK shares are considered compromised and hence the AKD_i distribute the new TEK shares via multicast and encrypted under individual root's child key with d child keys corresponding to the tree degree that are shared by the members (M_{ii}) in cluster i .i.e.

$$E[I_{\rho_i}] = d \quad (16)$$

Therefore we evaluate the total overhead induced by merging clusters i and v on member handoff as

$$O_{i,v}^{(m)} = RO_T + E[M_{i,v}] \quad (17)$$

We notice that the total overhead $O_{i,v}^{(m)}$ in (17) is equal to the expected number of re-key messages per unit time due to leave/join in cluster i and v respectively plus the impact overhead in terms of messages per unit time of cluster i on cluster v when merged.

Table 5 summarises the induced overheads on member handoff with consideration of both forward and backward secrecy while participating in s services.

TABLE 5
Overheads induced by member handoff with forward and backward secrecy

Over-heads	Rekeying approach	
	<i>Pairwise</i>	<i>LKH</i>
RO_T	$\lambda_i(\rho_i) + \lambda_v(\rho_v + s)$	$\lambda_i d \log_d(\rho_i) + \lambda_v(\log_d(\rho_v) + s)$
$E[M_{i,v}]$	$\lambda_i(\rho_v + s) + \lambda_v(\rho_i)$	$\lambda_i s + \lambda_v d$
$O_{i,v}^{(m)}$	$(\lambda_i + \lambda_v)(\rho_v + s + \rho_i)$	$\lambda_i(s + d \log_d(\rho_i)) + \lambda_v(d + \log_d(\rho_v) + s)$

TABLE 6
OVERHEADS INDUCED BY MEMBER HANDOFF WITH BACKWARD SECRECY ONLY

Over-heads	Rekeying approach	
	<i>Pairwise</i>	<i>LKH</i>
RO_T	$\lambda_v(\rho_v + s)$	$\lambda_v(\log_d(\rho_v) + s)$
$E[M_{i,v}]$	$\lambda_i(\rho_v + s)$	$\lambda_i s$
$O_{i,v}^{(m)}$	$(\lambda_i + \lambda_v)(\rho_v + s)$	$\lambda_i s + \lambda_v(\log_d(\rho_v) + s)$

However, due to cryptographically separate keys per cluster in SMGKM, we only consider member handoff with backward secrecy since handover members maintain session continuity at move hence making forward secrecy at the old cluster i redundant (i.e. $E[L_{\rho_i}] = 0$ and $E[I_{\rho_i}] = 0$ for both rekeying approaches). Table 6 summarises the overheads induced in SMGKM.

It can be observed from Table 5 that the convectional approaches induces high $E[M_{i,v}]$ due to 1-affect-n phenomenon. Thus when the TEK changes, the entire clusters undergo rekeying while in SMGKM, the $E[M_{i,v}]$ is optimized by performing rekeying only at the target cluster.

6 PERFORMANCE AND SIMULATION ANALYSIS

The performance of SMGKM scheme is analyzed through numerical analysis and simulations in terms of rekeying transmission overhead corresponding to the additional signalling load caused by rekeying, storage overhead corresponding to the storage capacity of the key management keys stored by the entities (M_i , AKD_i and DKD). The communication overheads for both rekeying approaches (*pairwise* and *LKH*) as a result of unicast or multicast transmissions of rekeying messages at the cluster level are also considered. Finally the security analyses section considers all types impossible attacks in SMGKM.

6.1 Rekeying Transmission overhead

Let the rekey signaling message delivery between the DKD and the AKDs be ω unit and between the MN and the AKDs be α unit respectively. The parameters ω and α are the weightings factors of the signal load at the core network and the wireless part of the framework respectively. They are used to determine link stability. Since the DKD locate the currently serving AKD_i which could be far away from the DKD then $\omega \gg \alpha$. Therefore by using the rekeying transmissions obtained in Table 5, the signalling cost induced by rekeying at the wired and wireless parts of the SMGKM when n -moves subscribed to s -multi-services occur can be formulated and compared to the convectional schemes as summarized in Table 7.

TABLE 7
COMPARISON OF REKEYING TRANSMISSION OVERHEADS

Reference framework		Transmission Overhead
DeCleene et al[13, 18]	BR	$10ns\omega + 6ns\alpha$
	IR	$3ns\omega + 5ns\alpha$
	FEDRP	$3ns\omega + 4ns\alpha$
GKMF[14]		$4ns\omega + 4ns\alpha$
Kellil et al[15]		$2ns\omega + 4ns\alpha$

SMGKM	$ns\omega + 4ns\alpha$
-------	------------------------

As shown in Fig.9, SMGKM outperforms the conventional protocols by giving minimum rekeying transmissions for increasing number of handoffs and services.

Fig.9. Rekeying Transmission overhead

By allowing the AKDs to perform rekeying process gives DKD scalability and reduces the signaling investment pressure at the core network. Therefore our system can be considered as signalling optimizer for efficient bandwidth utilization. Surprisingly IR outperforms GKMF because GKMF involve the DKD on every rekeying process which increases signaling at the core network for cumulative services. However, FEDRP, GKMF and Kellil et al give similar rekeying transmissions at the cluster level. Since the cluster level is bandwidth limited and subject to high packet loss, reduction in the rekeying transmissions depends on the rekeying approach used for efficient bandwidth utilization. Usually *LKH* rekeying approach [5] is favourable to reduce the rekeying transmission overheads at the cluster level as discussed in section 6.2

6.2 Rekeying communication overhead

In Table 8 and Table 9 we compare the communication overheads induced in SMGKM using *pairwise* and *LKH* rekeying approaches respectively at the cluster level compared to the conventional approaches. This requirement satisfies the bandwidth consumption of the bandwidth limited wireless system. Induction of high number of signalling messages transmitted (either by unicast or multicast) during rekeying process consumes enormous network bandwidth while contributing to delays in distributing the fresh service group communication keys.

TABLE 8
PAIRWISE REKEYING APPROACH COMMUNICATION OVERHEADS

Reference framework	Total Communication Overhead		Total C-Overhead
	Cluster i	Cluster v	
BR	$2 \times [O(n)+1+1]$	$s \times [O(n)+1+1+1]$	$2sO(n)+5s$
IR	$2 \times [O(n)]$	$s \times [O(n)+1]$	$2sO(n)+s$
FEDRP	0	$s \times [O(n)+1]$	$sO(n)+s$
GKMF	0	$s \times [O(n)+1]$	$sO(n)+s$
KELLIL	0	$s \times [O(n)+1]$	$sO(n)+s$
SMGKM	0	$O(n)+s$	$O(n)+s$

TABLE 9
LKH REKEYING APPROACH COMMUNICATION OVERHEADS

Reference framework	Total Communication Overhead
BR	$2sO(\log_a(n))+5s$
IR	$2sO(\log_a(n))+s$
FEDRP	$sO(\log_a(n))+s$
GKMF	$sO(\log_a(n))+s$
KELLIL	$sO(\log_a(n))+s$
SMGKM	$O(\log_a(n))+s$

It can clearly be observed from Fig.10 that using pairwise rekeying approach, BR induces high number of rekeying signalling messages than others because the service group keys and the local area keys are updated in both clusters during member handoff.

The IR reduces the need to rekey the service keys but triggers local area key rekey only at both clusters. To further reduce communication overheads from IR, both the GKMF and KELLIL et al schemes adopt DR strategy by introducing the use of mobility list as to record handover members such that the previous cluster i induces null communication overhead on handover. This actually improves the bandwidth efficiency of the system while satisfying the backward secrecy requirement of services at the target cluster v . However these protocols reduce communication overhead at the expense of storage complexity at the resource constraint mobile receiver as described in section 6.3. Thus handover members need to maintain the local area keys for the previously visited clusters to reduce the need for rekeying on return period. Whenever a handover member finally leave after visiting multiple clusters, rekeying is triggered in all the visited clusters hence inducing huge rekeying communication overheads hence enormous consumption of the system bandwidth.

Since data transmission cannot resume until all visited clusters have been rekeyed with new service keys ($TEK_{i,j}$), FEDRP, GKMF and Kellil et al stay offline longer than SMGKM.

Fig.10. Pairwise rekeying communication overheads

Fig.11 LKH rekeying communication overheads

Similarly, *LKH* rekeying tree based approach is the prominent method to reduce rekeying communication overhead from $O(n)$ to $O(\log_d(n))$ as depicted in Table 9. Using a full balanced *LKH* binary tree of degree $d=4$, It can be observed in Fig.11 that rekeying communication overhead for SMGKM still outperforms the conventional approaches in the presence of n -moves participating in s -services.

However for n multiple handoffs participating in s services, the convectional protocols incur huge communication overhead and huge delays in obtaining the services. If the service key shares require to be updated in conventional approaches, rekeying becomes a performance hurdle due to 1-affect- n phenomenon which disturbs the entire system hence high $E[M_{i,v}]$. These limitations has enforced the introduction of SMGKM scheme which is very adaptive multi-services with multi-handoffs. With cryptographically separate keys per cluster in SMGKM, the rekeying of the $TEK_{i,j}$ shares get localized hence alleviating 1-affect- n phenomenon. SMGKM also minimize the need to introduce easily compromised local area keys like in convectional schemes hence improving security as described in section 6.5 and storage efficiency at the handover member as shown in Fig. 12.

6.3 Storage overhead

Storage overhead is used to determine the number of keys held by the mobile receiver M_i , AKD_i and the DKD. This requirement enables fast execution and fast accessibility of stored keys if there are low keys held by the network entities. Table 10 present the storage overheads incurred by the network elements in the presence of n -moves participating in s -services.

TABLE 10
COMPARISON OF STORAGE OVERHEADS

Storage cost	DeCleene et al	GKMF	Kellil et al	SMGKM
At the DKD	s	$S+2$	s	n
At the DKD	$n+1$	$n+4$	$n+2$	$n+1$
At the member	$s+1$	$s+3$	$s+3$	3

From the two simulation scenarios in Fig.12, it can be seen that the conventional approaches add storage complexity to the resource limited member M_i by introducing

extra local area keys for localizing rekeying.

Fig. 12 Storage overhead for increasing n and s .

However the SMGKM adopt independent TEK per cluster without extra key memory required. Handover members revoke the cryptographic keys for the previously visited clusters after complete handoff hence improving the storage complexity. Therefore members leaving after visiting multiple clusters do not trigger repeated rekeying as in DeCleene et, GKMF and Kellil et al. Thus SMGKM incurs less storage overhead at the M_i , AKD_i and DKD than in convectional schemes in the presence of n -moves participating in s -services. Furthermore each $SKDL_i$ of AKD_i only maintains the rows for the concurrent active members they are currently serving hence serving the storage consumption. Handover members always maintain a set of 3 keys regardless of location which makes our system adaptive to multiple services compared to GKMF, DeCleene and Kellil et al schemes with dynamically changing storage cost at each entity. However both GKMF and Kellil et al induce high storage cost at the limited resource member due to introducing more encryption keys which may drain more battery compared to each TEK_i share which is light to process since $TEK_{i,j} \ll TEK_i$. Thus the sum the $TEK_{i,j}$ shares for N services is equivalent to 1 key in SMGKM.

6.4 Bandwidth consumption in SMGKM

In this section we explore the system bandwidth optimization in SMGKM with multiple services s compared to a single service case of the conventional protocols. Assuming a pool of handover members M_i in the system is equally likely to receive s services. Thus the average number of concurrent members per service s can be expressed as $\rho/s = \lambda/s\mu$.

Given the expected number of rekey messages per second for each service s , then (5) can be re-written as

$$E[R_{\lambda/s,\mu}] = \lambda/s(E[L_{\rho/s}] + E[J_{\rho/s}]) \quad (18)$$

Let B_{sys} and B_s define the system total bandwidth utilization and the multicast service download rate both in bits/s respectively. Let R_z define a constant rekey message size in bits. Clearly the total system bandwidth B_{sys} consumption can be obtained by

$$B_{sys} = sB_s + sR_z E[R_{\lambda/s,\mu}] \quad (19)$$

By taking in to account the membership dwell time in service s , we multiply (19) by $1/\mu$ time units such that

$$\frac{E[R_{\lambda/s, \mu}]}{\mu} = \frac{\rho}{s} (E[L_{\lambda_i/s, \mu_i}] + E[J_{\lambda_v/s, \mu_v}]) \quad (20)$$

$$\equiv F(s) \quad (21)$$

Therefore B_{sys} in (19) evaluate to

$$B_{sys} = sB_s + s\mu R_z F(s) \quad (22)$$

Now deviding (22) by B_s equally gives

$$\frac{B_{sys}}{B_s} = s + s\beta F(s) \quad (23)$$

where $\beta = \frac{\mu R_z}{B_s}$ is the dimensionless parameter.

It can be deduced from (23) that B_{sys} can be obtained if $E[L_{\rho_i/s}] + E[J_{\rho_v/s}]$ is known.

By using (11) and (13), the simulation result in Fig. 13 show that using *LKH* rekeying approach with a branching factor of $d=4$ gives the best approximation value of $E[L_{\rho_i/s}] + E[J_{\rho_v/s}] = RO_T$ for $s=1$ as proven in [5].

Fig.13 $E[L_{\rho_i/s}] + E[J_{\rho_v/s}]$ against for $d=2$ to 8

However in SMGKM, members handoff rekeying considers only backward secrecy hence making $E[L_{\rho_i/s}] = 0$. This optimises $F(s)$ (hence B_{sys}) given $E[J_{\rho_v/s}]$ at the target cluster v . If multiple handoffs occur while subscribed to s multiple services, $F(s)$ can rapidly increase due to considerable rekeying overhead caused by to join. This realistically induces high B_{sys} consumption. We now also optimize B_{sys} with respect to s or $E[J_{\rho_v/s}]$. To solve the SMGKM optimization problem, we express $F(s)$ in (21) in terms of $E[J_{\rho_v/s}]$ and s services subscribed by joining members ρ_v such that

$$F(s) = \frac{\rho_v}{s} (E[J_{\rho_v/s}]) \quad (24)$$

$$\approx \frac{\rho_v}{s} (\log_d(\frac{\rho_v}{s}) + s) \quad (25)$$

if we apply the LKH rekeying approach.

This reduces (22) to

$$\Psi(s) = s + s\beta \frac{\rho_v}{s} (\log_d(\frac{\rho_v}{s}) + s) \quad (26)$$

$$\text{where } \Psi(s) = \frac{B_{sys}}{B_s}.$$

Hence setting $\frac{\partial}{\partial s} (\Psi(s, \beta, \rho, d)) = 0$ for $s > 1$ gives the value of s^* that optimises $F(s)$, i.e.

$$s^* = \frac{\beta \rho_v}{(\beta \rho_v + 1) \ln(d)} \quad (27)$$

Now we explore the system bandwidth optimization in SMGKM by using the baseline system parameters of β , ρ_v , $1/\mu$ and data rate B_s defined in [23] for the same services illustrated in Fig.1. In our case the number of servers m represents the number of multicast services s provided by the SP. The estimated size of R_z is defined by IETF standard with at least rekey message header (HDR) field of size 28bytes, sequence number (SEQ) field of size 8 bytes, security association field of size 48bytes, key download payload (KD) field of size 11bytes in addition to the key size, signature (SIG) field of size 4 bytes and optional certificate (CERT) field. Thus R_z evaluate to 99 bytes in total. Since the $TEK_{i,j}$ shares are equivalent to the number of services, we choose R_z to be at most 2kB in our study including CERT field and the key size. The ratio of μ and R_z defining the parameter β in (23) has more contributing weight on the system bandwidth consumption which affect the overall system performance. Using the illustrative example in [23], it is found that β is likely to range relatively from 10^{-3} (stock quote services) to 10^{-6} (Internet TV/video services). Assuming a maximum of ten services provided and a total of $\rho_v = 10^5$ members under the same SG. If M_1 in Fig.1 perform handoff while participating in services s_1 (stock quote), s_2 (Internet radio), s_3 (one of the charged mobile Internet TV services like sports) with $\beta_1=10^{-4}$, $\beta_2=10^{-5}$, $\beta_3=10^{-6}$ respectively [23]. As explained in section 4.2 all the services $\{s_1, s_2, s_3\}$ subscribed by M_1 undergo rekeying for backward secrecy at rate R_z at cluster v including existing members M_v subscribed to same set of affected services.

The simulation result in Fig. 14 shows the ratio of multiple service approach to that of a single service approach.

Fig.14 $\Psi(s)/\Psi(1)$ versus s , given β for different services

It can be deduced that for some high values of β such as $\beta_1=10^{-4}$, as s increases in the network, $\Psi(s)/\Psi(1)$ decreases until it reaches minimum mostly due to reduction in rekeying overhead and may increase steadily due to increase in service download rate B_s . Hence there is optimal s^* to achieve optimized B_{sys} . Instead, for some low values of α such as $\alpha_2=10^{-6}$ and $\alpha_3=10^{-5}$. It can also be noticed that $\Psi(s)/\Psi(1)$ increases monotonically with cumulative services s mostly due to increase in service download rate compared to rekeying overhead. This proves that grouping a pool of members in to SGs can be optimal. However the bandwidth saved by rekeying cannot entirely alleviate the increase in extra multicast service traffic. For B_{sys} utiliza-

tion, we plot the optimal ratio of multicast service bandwidth to the total system bandwidth $\{s^*/\Psi(s^*)\}$ against β for various services in Fig. 15.

Fig. 15 Bandwidth utilization versus β .

It can be deduced that the bandwidth utilization for convectional schemes adopting single service approach decreases constantly. This is due to the fact that as β increases, either the multicast traffic download rate or members dwell time $1/\mu$ decreases hence a comparative increase in rekeying messaging. Surprisingly, the single service protocols achieve very low bandwidth utilization due to high rekeying overhead. Instead our SMGKM approach incorporating multiple service approach concurrently increases the utilization tremendously. This is because for large values of β (i.e. greater than the break-even point at $s^*=1$), rekeying overhead decreases to achieve high bandwidth utilization. Therefore splitting the services in to small service groups G_K could be adopted to reduce B_{sys} .

6.5 Security analysis

SMGKM uses cryptographically separate keys for each cluster. Therefore key compromise within the cluster does not affect other clusters. *Eavesdropping opportunities* are impossible in our system because backward secrecy requirement is achieved by updating the TEK_i shares for the affected services when a handover member rejoins the target cluster. *Impersonation attacks* are also intolerable due to mutual authentication between M_i and the AKD_i on every handoff. Each member is verified using the individually derived SK_{M_i,AKD_i} which is verified against the already derived SK_{M_i,AKD_i} by the trusted DKD in the $SKDL_i$ before accessing the fresh key shares for its services at each cluster. With mutual authentication over the air interface, M_i can check if a network can be trusted before entering. The AKD_i cannot forge to be legitimate because it is not aware of the M_i AK_{M_i} . It only receives the already derived SK_{M_i,AKD_i} for M_i from the trusted DKD. *Anonymity* can be attained by utilizing temporary identities to identify the network nodes and encrypt these identities over the air interface. To prevent *traffic analysis attacks*, an identity is transmitted encrypted in unique SK_{M_i,AKD_i} per cluster i thus prevent eavesdropping and identification of active terminals. Each row of the $SKDL_i$ held by the AKD_i is integrity protected using MAC to prevent possible *replay attacks* on the information stored. This is vital for securely storing accounting and location information at cluster level instead of periodically updat-

ing the DKD for further processing. *Physical node capture attacks* cannot expose our system security keys to compromise due to secure verification of key hierarchy derivation steps considered in SMGKM. The SK_{M_i,AKD_i} in $SKDL_i$ are maintained in encrypted form together with the MAC that provides integrity protection and this provide authentication of each $SKDL_i$ row. If the $SKDL_i$ rows encryption/decryption, verification, key derivation and integrity protection procedures are implemented in an external secure hardware chip like the trusted module, it is practically more infeasible for the adversary to obtain the SK_{M_i,AKD_i} by just examining the AKD_i $SKDL_i$ or between the AKD_i - AKD_i links. The $SKDL_i$ concept also inhibits *location based attacks* due the encrypted rows thus *redirection attacks* are also impossible.

7 CONCLUSION

In this paper, a new SMGKM scheme has been proposed to improve the key management performance in the presence of multi-moves participating in multi-group services. It considered providing backward confidentiality where mobile receivers dynamically perform handoff while seamlessly maintaining diverse subscriptions. In contrast to convectional schemes targeted for a single service, SMGKM used a new rekeying strategy based on lightweight KUS and SKDL for effectively performing key management and authentication phases respectively during handoff. SMGKM adopted independent TEK per cluster to localize rekeying and mitigate 1-affect-n phenomenon. By offloading the key management and authentication phases to the intermediate AKDs massively reduced signalling load at the core network than in convectional schemes hence giving DKD scalability while preventing bottlenecks. The SMGKM analytical model was developed for two rekeying approaches: *pairwise* and *LKH*. Numerical analysis and simulation results of the SMGKM performed much better using both rekeying approaches in comparison to previous work. Thus SMGKM have shown significant resource economy in terms of communication bandwidth overhead, storage overheads at the DKD, AKD, and the mobile receiver while providing intense security. Finally, the analytical study was explored by simulation for solving the bandwidth optimization problem in SMGKM which showed efficiency in bandwidth consumption in the presence of multi-services. However, SMGKM is expected to become a practical dynamic solution for securely and efficiently managing multi-services which can be received concurrently by huge mobile subscribers in the future wireless networks such as emerging Software-Defined Networks.

ACKNOWLEDGMENT

This work was supported partially from TSB UK under grant application KTP008734 and MoESDP-DTEF (Ministry of Education Skills & Development Planning, Department of Tertiary Education and Financing), Gaborone, Botswana. The authors are also very grateful to the reviewers comments which immensely improved this work.

REFERENCES

- [1] G. S. V. R. K. Rao and G. Radhamani, *WiMax: A Wireless Technology Revolution*: Auerbach Publishers, Incorporated, 2008.
- [2] 3GPP, "Multimedia Broadcast/Multicast Service; Stage 1 (Release 8)," *Technical Specification 3GPP TS 22.146* vol. 8.3.0, (2007-06), June 2007.
- [3] 3GPP, "Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); LTE; Multimedia Broadcast/Multicast Service (MBMS); Stage 1 (Release 9)," *Technical Specification 3GPP TS 22.146* vol. 9.0.0, (2010-01) 2010.
- [4] P. Judge and M. Ammar, "Security issues and solutions in multicast content distribution: a survey," *Network, IEEE*, vol. 17, pp. 30-36, 2003.
- [5] W. Chung Kei, M. Gouda, and S. S. Lam, "Secure group communications using key graphs," *Networking, IEEE/ACM Transactions on*, vol. 8, pp. 16-30, 2000.
- [6] S. Rafaeeli and D. Hutchison, "A Survey of Key Management for Secure Group Communication," *ACM Computing Surveys*, vol. 35, pp. 309-329, September 2003.
- [7] T. T. Mapoka, "Group Key Management Protocols for Secure Mobile Multicast Communication: A Comprehensive Survey," *International Journal of Computer Applications*, vol. 84, pp. 28-38, December 2013.
- [8] T. T. Mapoka, S. Shepherd, R. Abd-Alhameed, and K. Anoh, "Efficient Authenticated Multi-Service Group Key Management for Secure wireless mobile multicast)" in *IEEE Third International conference on Future Generation Communication Technologies (FGCT 2014)*, Luton, UK, 2014, submitted for publication. (Pending publication).
- [9] Y. Challal and H. Seba, "Group Key Management Protocols: A Novel Taxonomy.," *Enformatika, International Journal of Information technology*, vol. 2, 2005.
- [10] H. Harney and C. Muckenhirn, "Group Key Management Protocol (GKMP) Specification," RFC 2094, July 1997.
- [11] T. Hardjono, B. Cain, and I. Monga, "Intra-domain Group Key Management for Multicast Security," *IETF Internet draft*, September 2000.
- [12] S. Mittra, "Iolus: a framework for scalable secure multicasting," *SIGCOMM Comput. Commun. Rev.*, vol. 27, pp. 277-288, 1997.
- [13] B. DeCleene, L. Dondeti, S. Griffin, T. Hardjono, D. Kiwior, J. Kurose, et al., "Secure group communications for wireless networks," in *Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. IEEE*, 2001, pp. 113-117 vol.1.
- [14] 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)*, 2007, pp. 100-107.
- [15] M. Kellil, Olivereau, J. C. A., and P. Janneteau, "Rekeying in secure mobile multicast communications," *United States Patent Application Publications*, US 2007/0143600 A1 2007.
- [16] S. Gharout, A. Bouabdallah, M. Kellil, and Y. Challal, "Key management with host mobility in dynamic groups," presented at the Proceedings of the 3rd international conference on Security of information and networks, Taganrog, Rostov-on-Don, Russian Federation, 2010.
- [17] M. L. M. Kiah and K. M. Martin., "Host Mobility Protocol for Secure Group Communication in Wireless Mobile Environments," *International Journal of Security and its Applications*, vol. 2, pp. 39-52, January 2008.
- [18] C. Zhang, B. DeCleene, J. Kurose, and D. Towsley, "Comparison of inter-area rekeying algorithms for secure wireless group communications," *Performance Evaluation*, vol. 49, pp. 1-20, 9/2002.
- [19] J. Loughney, M. Nakhjiri, C. Perkin, and R. Koodli, "Context Transfer Protocol (CXTP)," *IETF RFC 4067*, 2005.
- [20] S. K. and S. Frankel, "RFC4868: Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec " *IETF RFC 4868*, May 2007.
- [21] K. C. Almeroth and M. H. Ammar, "Collecting and Modeling the Join/Leave Behavior of Multicast Group Members in the MBone," presented at the Proceedings of the 5th IEEE International Symposium on High Performance Distributed Computing, 1996.
- [22] K. C. Almeroth and M. H. Ammar, "Multicast group behavior in the Internet's multicast backbone (MBone)," *Communications Magazine, IEEE*, vol. 35, pp. 124-129, 1997.
- [23] C. Kin-Ching and S. H. G. Chan, "Distributed servers approach for large-scale secure multicast," *Selected Areas in Communications, IEEE Journal on*, vol. 20, pp. 1500-1510, 2002.
- [24] L. Kleinrock, *Queueing Systems. John Wiley & Sons*, 1975.
- [25] D. Wallner, E. Harder, and R. Agee, "Key Management for Multicast : Issues and Architecture," National Security Agency, RFC 2627, June 1999.

Trust T. Mapoka received BEng(hons) and MEng degrees in Telecommunications & Internet Engineering from the University of Bradford, UK, in 2010 and 2011 respectively. He is currently enrolled as a PhD researcher also from the school of Engineering and Informatics at the same institution. His research interests include Multi-User Multi-Service (MUMS) security for highly dynamic mobile multicast communications, mobility management, network security, cryptography and information privacy, Access control ,Group Key management for secure group communication in future wireless networks. He is a member of IEEE.

Simon Shepherd is Professor of Computational Mathematics in the University of Bradford. He received the B.Eng degree with First Class Honours from the Royal Naval Engineering College Manadon in 1986 and the PhD degree in signal processing from the University of Plymouth in 1993. He is a Member of the London Mathematical Society, a Fellow of the Institute of Mathematics and a Fellow of the Royal Society of Medicine. He has published over 70 international journal papers and over 50 conference contributions, a book and several patents. His current research interests include cryptography, computer communications security and the applications of advanced techniques and algorithms in digital signal processing to pressing medical problems.

Raed A Abd-Alhameed is Professor of Electromagnetic and Radio Frequency Engineering at the University of Bradford, UK. He has long years' research experience in the areas of Radio Frequency, Signal Processing, propagations, antennas and electromagnetic computational techniques, and has published over 400 academic journal and conference papers; in addition he is co-authors of three books and

several book chapters. At the present he is the leader of Radio Frequency, Propagation, sensor design and Signal Processing; in addition to leading the Communications research group for years within the School of Engineering and Informatics, Bradford University, UK. He is Principal Investigator for several funded applications to EPSRCs and leader of for several successful knowledge Transfer Programmes such as with Pace plc, YW plc, ITEG Ltd, Datong plc, Emkay Ltd, and Two limited. He has also been a co-investigator in several funded research projects including: 1) Nonlinear and demodulation mechanisms in biological tissue (Dept. of Health, Mobile Telecommunications & Health Research Programme and 2) Assessment of the Potential Direct Effects of Cellular Phones on the Nervous System (EU: collaboration with 6 other major research organisations across Europe). Recently he was awarded the business Innovation Award for his successful KTP with Pace on the design and implementation of MIMO sensor systems. He is the chair of several successful workshops on Energy Efficient and Reconfigurable Transceivers (EERT): Approach towards Energy Conservation and CO₂ Reduction that addresses the biggest challenges for the future wireless systems. He has also appointed as guest editor for the IET Science, Measurements and Technology Journal since 2009, and 2012. He is also a research visitor for Wrexham University, Wales since Sept 2009 covering the wireless and communications research areas. His interest in computational methods and optimisations, wireless and Mobile communications, sensor design, EMC, beam steering antennas, Energy efficient PAs, RF predistorter design applications. He is the Fellow of the Institution of Engineering and Technology, Fellow of Higher Education Academy and a Chartered Engineer.