Multimed Tools Appl (2015) 74:2899–2912
DOI 10.1007/s11042-013-1753-9

Huffman based conditional access system for key distribution in digital TV multicast

R. Varalakshmi · V. Rhymend Uthariaraj

Published online: 14 November 2013

Abstract The advance of modern network technologies has made Digital TV systems available throughout the world. To provide secure media delivery in Digital TV systems, a large number of messages are exchanged for key updates in the conventional key distributed schemes of conditional access systems (CAS). In this paper, proposed a new multicast key distribution scheme based on Huffman grouping scheme of access control for conditional access system (CAS) in digital television multicast. The proposed key distribution scheme can greatly reduce the computation using fast Fourier transform and acquire higher efficiency and security using extended Euclidean algorithm. With this scheme, only authorized subscribers can watch the subscriber programs correctly. Unauthorized subscribers have no information to retrieve the correct programs over the networks. Moreover, the proposed scheme is more flexible in processing subscribers joining and leaving which is achieved by using Huffman based grouping scheme and is very important for service provider to dynamic manage the subscriber.

Keywords Multicast key distribution scheme · Huffman grouping scheme · Conditional access system · Digital TV multicast · Fast Fourier transform · Extended Euclidean algorithm

1 Introduction

Multicasting is a digital television technology that gives viewers access to additional local multicast TV programs. A single station can now provide multiple programs of separate programming simultaneously, free and over the air. Each separate program stream is called a multicast. Digital TV Multicast is done by using a single digital transmitter to send different programs simultaneously. Groups/subscribers can decide which of these programs to watch. To provide secure media delivery in Digital TV systems, a large number of messages are exchanged for key updates in the conventional key distributed schemes of conditional access systems (CAS).

To ensure access rights of the authorized subscribers who pay for the content watched and prevent media/video programs from unauthorized access, scramble and encryption algorithms

R. Varalakshmi (✉) · V. R. Uthariaraj
Ramanujan Computing Centre, Anna University, Chennai, India
e-mail: rvaralakshmi697@gmail.com

are commonly used for secure media delivery and channel protection. The encryption keys should be distributed to all subscribers so that they can receive and decrypt the subscribed video programs or media streams. For large amounts of subscribers in a conditional access system, traditional key distribution schemes [2–4, 8–10, 14, 19–21] result in high computational costs and poor quality of service. To provide real-time video services, an efficient and secure key distribution scheme is a necessary and important requirement.

The basic components in a CAS [5] in Digital TV system are a service provider (SP) and large amount of subscribers. Before receiving video programs from the service provider, a subscriber must first register with the service provider and get his own secret key and along with other secret information. A scrambling/descrambling function is usually used for channel protection in CAS. The scramble keys/control words (CW) initialize the generation of the pseudo-random sequence. The descramble function recovers the original video programs at the receivers with the help of the CW and the pseudo-random sequence. For CAS security, CW will be changed once per 5–20 s. A superior CAS [5] should be of high security, efficiency in processing stream and flexible in dynamic management.

In this paper, a new key distribution for CAS based on Huffman grouping scheme for access control has been introduced. This scheme is secure enough. Moreover, the proposed scheme is more flexible in processing subscribers joining and leaving which is achieved by using Huffman based grouping scheme and is very important for service provider to dynamic manage the subscriber. This paper is organized as follows. In Section 2, various key distribution schemes for program protection are discussed. Section 3 describes Huffman based grouping scheme for key distribution that can periodically update the encryption keys of the CWs for subscribed programs. Section 4 describes the performance analysis of the proposed scheme in comparison with previous schemes. Section 5 gives the Simulation Results. Finally, the conclusions are given in Section 6.

2 Related work

J.W. Lee proposed a key distribution scheme [8] for subscription channels. A four-level key hierarchy is used in Lee's scheme: CW, Direct Entitlement Key (DEK), DK, and Master Private Key (MPK). CW and DEK perform the same functions as the CW and AK in the ITU recommendation, respectively. The DK consists of a Private Key (PK) and a Group Key (GK) and is used to encrypt the DEK. PK is used uniquely for each subscriber and GK is used as a group key for each group of channels. MPK is used to encrypt the entitlement management message and DK is stored in a smart card-based device. Keys of the last three levels are never revealed outside the smartcard-based device, which the CW is sent out to descramble the subscribed programs. The computation of encrypting and broadcasting keys in Lee's scheme are too heavy to provide PPV services. In addition, the PK is unnecessary, since MPK can be used to identify the subscriber.

To improve Lee's scheme, Tu et al. [19] modified DEK in Lee's scheme and replace DK with a newly proposed key Receiving Group Key (RGK). The RGK is used for subscription channels only. Subscribers whose authorization is expired will not receive the new RGK and can no longer vie video programs. In Tu's scheme, all subscribers are classified into charging and receiving groups, where subscribers with the same charging periods are put in the same charging group and subscribers with the same set of subscription channels in the same receiving groups. Tu's scheme is efficient for subscription management and has the advantages to distribute the heavy daily work. However, the maximum number of the receiving group becomes the total number of subscription channels and is still a very large number. Besides,

DEK update in Tu's scheme require large amount of package broadcasting. In summary, the aforementioned schemes may introduce high computation and transmission cost for key updates. This is inefficient and costly when the client side is using a smart card with low computing power.

Akl and Taylor were the first authors to propose a top-down structure for solving the hierarchical access control problem [1]. Moreover, comparing with Akl's scheme, L. Harn's [6] scheme is more efficient in the memory usage since it needs less space to keep the public information. As a review of Harn's scheme [6], T. Jiang et al. [7] proposed a hierarchical structure for conditional access system. It adopts four level key hierarchy that is CW, AK, RGK and MPK. Here they divide the program channels into several channel groups and every group needs only one AK to encrypt the CWs of all the channels in the group to form ECM package. InT. Jiang et al., channel groups are formed based on some combinational channel groups, there is well-defined combination group given. If the combination grouping scheme is more simple, the conditional access system (CAS) will be more convenient and efficient in key distribution and management. For simplifying grouping of the program channel, we proposed the Huffman based grouping scheme access control [17, 18]for conditional access system (CAS). Moreover, the proposed scheme is more flexible in processing subscribers joining and leaving which is achieved by using Huffman based grouping scheme and is very important for service provider to dynamic manage the subscriber. Furthermore, the proposed system is well-suited for the DTV standard, which can be used for both PPC and PPV services.

3 Key distribution based on Huffman grouping scheme for CAS

In DTV multicast, a service provider normally provides many DTV programs and the subscriber can subscribe their favoriteprograms. Some subscribers may subscribe more programs including all the programs which are subscribed by other subscribers. That is, subscribers who subscribe more programs have higher privilege than those subscribe few programs. This can be considered as a cryptographic solution by using hierarchical access control relation as in [2, 3, 6, 10, 12, 21]. So we adopt a customized Huffman grouping scheme in programs for CAS, which can get a good efficiency in reducing the computation of encryption and quantity of message being transferred for key distribution as well as flexibility in dynamic management.

In this proposed scheme as in [17, 18], we still adopt four level key hierarchy, that is CW_i, AK_i, RGK_i and MPK_i. We divide the programs into several groups and every group needs only one AK_i to encrypt the CW_is of all the programs in the group to form ECM package. For example, for one DTV multicast station, there are 100 DTV programs. We divide these programs into four basic program groups as shown in Fig. 1. That is movie program group G1, stock information program group G2, news program G3, and sports program group G4, where chi denotes the i^{th} program. Based on the probability of transmission, we derive several combinationalgroups using customized Huffman grouping scheme. The Huffman technique requires probability of transmission for each group as input and output will be the variable length code in binary for each of the groups in such a way that the group having the higher probability of transmission will have lesser length codes. The Huffman technique is modified in such a way that it generates the GID for each group. We need least number of bits for the group leaving in the more recent past and more bits for the one who leaves in the less recent past (or the more distant past). The problem of GID generations is now mapped to a problem of variable length code generation.

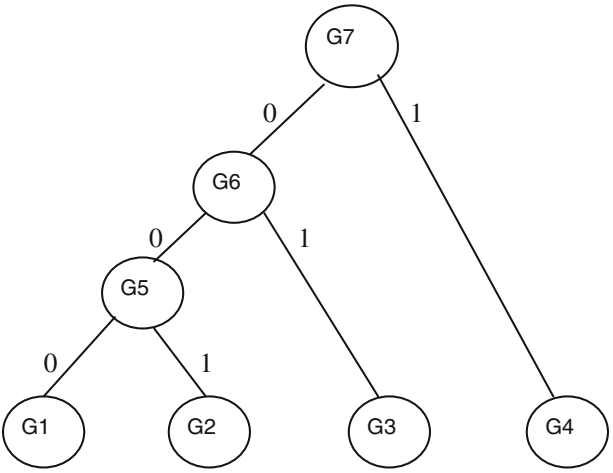

Fig. 1 Huffman grouping scheme for CAS

However the Huffman's technique has to be customized in such a way that it takes the duration of stay as the input instead of the probability of transmission.

Steps involved in generation of GroupID

- The authentication agent calculates the expiry of the subscription from the subscription fee paid by the group.
- The duration of stay has to be mapped in such a way it can be plugged as an input to the customized Huffman's grouping scheme instead of probability of transmission.
- Use Customized Huffman's grouping scheme to get the ID for each of the Group.
- The Group IDs generated by this scheme is related to the keys possessed by them.

This method of ID generation makes sure that the group with expiry of subscription in near future has the least number of bits in its ID. S0 the number of keys that has to be changed when that group leaves the group is going to be less. However because of variable length coding there are going to be groups whose GIDs have more bits than what they would have had if their GIDs were represented by normal binary notation. By reconstructing these GIDs again by this approach after some specific time period this can be resolved. This is because these groups subscription expiration is now closer to the expiration deadline then previously. We call the basic program group as leafnode and the combinational group above leaf node as mid-nodesuch as G5,G6 and the highest privilege group G7 as rootnode. The subscriber of higher privilege node in the Huffman basedaccess control grouping scheme can access the program that can beaccessed by subscriber of the lower privilege node which issubordinated to the higher privilege node. For example, subscribers in group G5 can access both the programs which can be accessed by the subscriber in group G1 and G2, i.e. movie program and stock information program from program 1 to program50. Subscriber in G7 can access all the programs provided by the service provider.

3.1 Generation of GID

The generation of GID can be illustrated with an example. If the duration of stay of the group is calculated as 5,10,20,25 time units then the GroupIDs can be generated as follows.

The time units are mapped to an input, equivalent to the probability of transmission. Each subscribers validity period is divided by the sum of duration of stay of all the groups. This gives the values 5/60, 10/60,20/60,25/60. The inputs for generation of GID for each of the subscribers are assigned in such a way that the subscriber with least time units of validity has the highest value and henceforth. Groupl 25/60, Group2 20/60,Group3 10/60,Group4 5/60.

This can be used converted to Huffman's codes represented as in Table 1.

So by having less number of bits for the group leaving most recently we can reduce the number of keys to be changed after the leave. Due to the dynamic nature of the group, and the possible expiration of keying material, it is necessary to update both the MPK and RGKs using rekeying messages. The Four operations involved are Key Refreshing, system Initialization, key updating when a new user joins the service, and key updating when a user departs the service. In the discussions that follow, we use an integer-valued time index to denote the time intervals during which fundamental operations occur, and assume that there is a system-level mechanism that flags or synchronizes the users to the same time frame. We shall always use the time index to denote the interval for which the new key information will become valid. Time interval will correspond to the time interval during which the new key information is being transmitted. Further, time interval corresponds to the interval of time during which a new member contacts the service provider wishing to join, or a current member announces to the service provider his desire to depart the service. We have depicted these cases in Fig. 2. Observe that it is not necessary that the time intervals have the same duration.

3.2 Key refreshing

Refreshing the session key is important in secure communication. As a session key is used, more information is released to an adversary, which increases the chance that a MPK will be compromised. Therefore, periodic renewal of the session key is required in order to maintain a desired level of content protection. By renewing keying material in a secure manner, the effects of a session key compromise may be localized to a short period of data. Algorithm 1 shows the Key refreshing algorithm that is used to refresh the key in secure communication. The crypto-period associated with a session key is governed by many application-specific considerations.

Algorithm 1: Key Refreshing algorithm

```
Step 1:  RGK(t+1)=RGK(t-1) // Since the amount of data encrypted using RGKs is usually much
                           // smaller than the amount of data encrypted by a session key, it is not
                           // necessary torefresh RGKs. Therefore, RGKs from the previous time
                           // interval (t-1) carry over to the next time interval.
Step 2:  α_g(t)= E(RGK(t), MPK(t)) // Rekeying message sent to users.
                           // Update the session key MPK(t-1) to a new session key MPK(t).
```

3.3 Group key distribution

The keys for each group are distributed as follows:

Step 1. Initially, the System selects a large prime number p and q, where p>q and $q \leq$p/4. The value p helps in defining a multiplicative group z_p^* and q is used to fix a

Table 1 Huffman Code representation

Symbol	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6	Group 7
Probability	5/60	10/60	20/60	25/60	15/60	35/60	1
Huffman Codeword	000	001	01	1	00	0	–

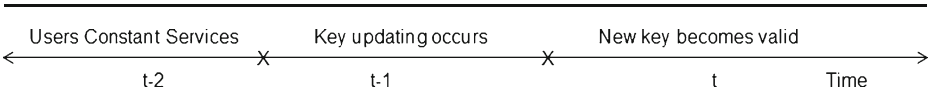

Fig. 2 Time intervals

threshold value δ, where δ=a+q. The value 'a' is a random element from the group z_p^* and hence when the 'a' value increases, the value of δ also increases. System selects a random element β from z_p^*.

Step 2. System now computes the shared secret key $RGK_i = \beta^{q} \bmod p$.

Step 3. For each group, system calculates $L = \prod AK_i$.

Step 4. The system computes a HCF value of $f^1(\delta, L)$ by using the extended Euclidian algorithm described in [11, 18] from which it finds x, y, b such that $x \times \delta + y \times L = b$. Then, the system multicasts RGK_i, x, p, q and d to the group members.

Upon receiving all the above information (β, x, p, q, b) from the system, an authorized subscriber u_i of the current group executes the following steps to obtain the new group key RGK_i .

Step 1. Computes MPK_i using the relation $x \bmod AK_i = MPK_i$.

Step 2. Computes δ using $MPK_i^{-1} \bmod AK_i = \delta$.

Step 3. Performs the following operation to find the shared secret key.

$$\beta^{b\times\delta} \Big/ \beta^{q} \bmod p = \beta^{(b\times\delta)-q} \bmod p = RGK_i.$$

The RGK_i obtained in this way must be equal to the RGK_i computed in Step 2 of system.

So in this scheme, if a subscriber subscribes some kind of the program, the system only needs to send a RGK_i of the program group to him. In order to keep flexible to process the subscriber's joining and leaving and reduce the load of key distribution for CAS, distributing RGK should be flexible and less computation or encryption. In this proposed Huffman based grouping scheme as in [17, 18], for each group, CAS only needs to distribute one encrypted message, that is $E(RGK_i)$, which is usually contained in EMM. All the subscribers in the group can use their MPK_i to decrypt the message to get RGK_i. Moreover, this Huffman based grouping scheme is more flexible in processing the joining and leaving.

3.4 Subscriber join

When a subscriber i want to join the group, system only needs to choose a unique key β and computes MPK_i for the subscriber. At the same time, system use the key βas a part of the encrypting key to encrypt the RGK of the subscribed group and send the MPK_i by smartcard to the subscriber.

3.5 Removal from the group due to expiry of subscription

If the expected subscriber leaves the group, that is the subscriber leaves according to the expiry of the subscription, then the system has to encrypt the new Session key with $AK_i + 1$, $AK_i + I'$ first, where 'i' is the most significant bit number of the subscriber. In other words, if the leaving subscriber has 3 bits gid then first step of encryption is with AK3 and AK3′. This encryption in algorithm 2makes sure that every subscriber with one or more bits more than the leaving subscriber will get the new group key. Then for the subscribers with same number of

bits, the compliments of each of the bits of the leaving subscriber starting from the least significant bit is encrypted and it is repeated ith next higher order bits until all the subscribers of the same length GID gets the Group Key.

Algorithm2: Encryption algorithm for expiry of subscription

```
Encrypt the new group key MPK_i with AK_i+ l, AK_i+ I' ('i' length of leaving subscriber's GID)
If there are some subscribers with same length GID in the group  j=0;
do
        encrypt with jth bit's compliment of leaving subscriber's GID
j=j+l;
while(all the subscribers of same length GID got the group key);
else
Stop the encryption.
```

Since Huffman grouping scheme is more simple, the CAS will be more convenient and efficient in key distribution and management. In this scheme, AK refreshed based on Huffman grouping scheme is used. Let us consider the case when user un-departs the group at timeframe t-1. Since user unknowns β(t-1) and RGK(t-1) these keys must be renewed. First RGK_i is renewed. Next, the session key β(t) is updated. The System forms a new β(t) and encrypts using the new $RGK_i(t)$ to form α(t) = E(RGKi(t), β(t)). This message is then sent to the subscribers. Because of small quantity of transferred message for rekeying, system can periodically transfer EMM in a short time to ensure that the newly subscribers can receive it in time. All of the process for rekeying can be done online which is more flexible and important for CAS. This Huffman based grouping scheme is discussed above mainly for pay-per-view (PPV), where period of rekeying for RGK depends on the specific program's lasting time and only one AK_i for each node is used.

3.6 Security analysis

Computing the newly updated RGK_i in the proposed scheme depends on the method used to calculate the subscriber's secret key AK_i in a specific amount of time. In this scheme, the group center distributes the parameters(β, x, p, q, b) to the subscribers through Huffman based grouping scheme based multicast communication. Hence an intruder can try to capture all the distributed parameters as well as and the threshold value δ. This δ can be computed only by using the subscribers secret key AK_i.

In this scheme, if the intruder is not an active adversary, it can use brute force attack. If the size of AK_i is ω bits, then the intruder has to use the total number of trials of 2ω. In this proposed work, the size of AK_i must be 64 bits. If the time required to perform one attempt using brute force attack is 1 μs, then the total time required will be 263 μs. Therefore when a large size AK_i is used, it is not possible to find the value of δ and hence RGK_i can't be computed by an adversary. Section 3.6.1 shows the security proof about the group key distribution.

3.6.1 Security proof

Given that $\delta < AK_i$, i=1…n and with every AK_i prime (or coprime at least), it is clear that:

$$\mathrm{HCF}(\delta, AK_i) = 1, \text{for every } i = 1, \ldots, n \tag{1}$$

and hence,

$$\mathrm{HCF}(\delta, L) = 1. \tag{2}$$

Equation (2) ensures, by the Extended Euclidean Algorithm, that the existence of x, y ∈ z_p^* such that $x\times\delta+y\times L=1$, from where it is deduced that $\delta.x \equiv AK_i^{-1}$ and so $x^{-1} \equiv AK_i\ \delta$, for every i=1,…,n. The Chinese Remainder Theorem guarantees that the solution for x^{-1} mod $AK_i = \delta$ and $\delta < AK_i$, for every i=1,…,n is unique.

The value $RGK_i = \beta^q \bmod p$ is obtained as shown next:

$$\begin{aligned}\beta^{\delta} &\equiv_p \beta^{a+q}\\ &\equiv_p \beta^{a}.1\\ &\equiv_p \beta^{a}\end{aligned} \tag{3}$$

β is public, but the use of δ assures that an outsider will not be able to guess 'a' since the value 'a' is any random element from the group z_p^*, and therefore, RGK_i. New values for p, β, q and/or a must be chosen for each refreshment of public and private key pairs. Note that δ, x, y depend on them and will change as they do. For security reasons, the service provider might decide to refresh the public and private keys after a long period of time with no members joining or leaving. Therefore, all of these keys are secured for group key distribution.

4 Performance evaluation

Assumption Let $L = \prod_{i=1}^{n} AK_i$ be a multiplication function which is used for member join operation, where AK_i = secret is the key of a user. Now, 'σi' is the size of the AK_i, where *i*=1, 2, 3, . . ., n (*n* = size of the group).

For optimizing the number of multiplication operations used for computing there exist faster multiplication algorithms, based on the fast Fourier transform, a divide and conquer approach [13, 15, 16] is used in this proposed key distribution algorithm. The idea is : multiplying two numbers represented as digit strings is virtually the same as computing the convolution of those two digit strings. Instead of computing a convolution, one can instead first compute the discrete Fourier transforms, multiply them entry by entry, and then compute the inverse Fourier transform of the result. Based on this idea, the number of multiplication operations to be performed in total to obtain the solution for the 'σ' digit number will be O σ (n log n). The Fast Fourier Transform (FFT) is a way to compute the Fourier transform of a sequence A in time O(n logn) instead of $O(n^2)$ with the classical way, when n is a power of 2.

Therefore it is faster than the traditional multiplication, which requires σ^2 single-digit products and the complexity is $O(\sigma \log_2 4)$. The fast Fourier transform multiplication approach works well when the value of σ>4,000 digits. However, if the number of digits of σ<16, this algorithm shall not show a significant difference. In order to optimize the use of the fast Fourier transform multiplication approach [13, 15, 16], the group size in this proposed key distribution algorithm can have 16-digits, 32-digits, 64-digits, 128-digits, etc. In the proposed algorithm, the analysis and testing of the algorithm for a group size p as 16-digit, 32-digit and 64-digit prime numbers. The key values used in the algorithm are 16 and 32 digit numbers.

Theorem 1 The number of multiplications in the computation of L is in the order of O(ω log (ω)) when fast fourier transform divide and conquer multiplication is employed for the key computation process where the key size is a n digit number.

Proof Two integers A and B, of length n represented as a polynomial in base x. It is important to stress at this stage that the length n has to be a power of two (n is even). In the

implementation there will be some processing to ensure that n is always even.

$$A(x) = A_0 + A_{1x} + A_2x^2 + \ldots\ldots + A_{n-1}x^{n-1}$$
$$B(x) = B_0 + B_{1x} + B_2x^2 + \ldots\ldots + B_{n-1}x^{n-1}$$

Split A, and B in the following manner:

$$A_0 = a_0 + a_2 + \ldots\ldots + a_{n-2} \text{ and } A_1 = a_1 + a_3 + \ldots\ldots + a_{n-1}$$
$$B_0 = b_0 + b_2 + \ldots\ldots + b_{n-2} \text{ and } B_1 = b_1 + b_3 + \ldots\ldots + b_{n-1}$$

Both halves are equal to $\omega/2$ as the length n is a power of two. However, one half possesses the even positions and the other the odd positions. Given a sequence $A = (a_0, a_2, \ldots, a_{\omega-2})$, compute its Fourier transform according to the formulae

Choosing for ω_k the complex roots of unity

(Note: i is equal to −1)

$$W_k = \exp\left(\frac{2ik\pi}{2n}\right) = \omega^k, \omega = \exp\left(\frac{2i\pi}{2n}\right)$$

This formula is used to compute the complex roots of unity needed to evaluate the number at 2n distinct points.

This would give us:

$$F_{2n}(A) = (c_0, c_1, \ldots, c_{2n-1}), b_k = \sum_{j=0}^{2n-1} a_j.\omega^{jk}$$

From above it gets the sequence $F_{2n}(B) = (d_0, d_1, \ldots, d_{2n-1})$,

The two sequences are multiplied together and produce the third sequence E

$$E = (e_0, e_1, \ldots, e_{2n-1}), \text{where } e_k = a_k * b_k$$

On sequence E, use the inverse Fourier Transform. This produces the sequence G

$$G = (g_0, g_1, \ldots, g_{2n-1}), g_k = \sum_{j=0}^{2n-1} e_j.\omega^{-jk}$$

Finally, dividing each of the resulting integers by 2n will give us the coefficients that construct the product of the multiplication.

$$\text{Time Complexity of FFT} = T(\omega)$$
$$T(\omega) = 2T\left(\omega/2\right) + O(\omega) = O(\omega \log \omega).$$

5 Simulation results

The proposed method has been simulated in NS-2 for more than 500 users and it is found that the computation and communication time with existing approaches to perform the rekeying operations.

To investigate the recital assessment of the Customized Huffman technique over normal Boolean logic minimization, the number of keys until the point of restructuring of the network is plotted. Figure 3 depicts the Highly stable Network where the user leave is predictable.

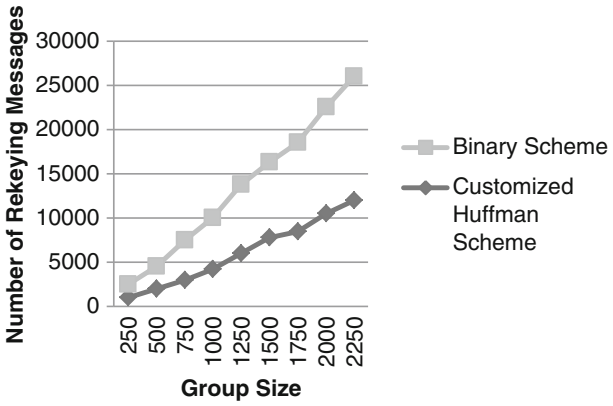

Fig. 3 Recital assessment for highly stable network

Figure 4 depicts The Network where the users leave is predictable with a good probability and Fig. 5 illustrates the Network where the users leave is highly unpredictable. The network has to be restructured when the recital assessment of Huffman technique goes below the Boolean logic minimization technique. The way in which this is simulated is through constructing the Huffman code and binary GroupIDs for all the users in the network and simulating the user leaving the network.

Highly stable Network where the user leave is predictable

The steps for simulation.

- Generate the binary notation user Ids for all the users.
- Generate the Group Ids using the Customized Huffman technique.
- The user leave is predictable in this Network. Always remove the predicted user from the network.
- Compute the number of re-keying operations.
- The algorithm proposed in the scheme was used for computing the number of rekeying for the Customized Huffman Scheme.
- Plot a graph between these two data.˘

The Network where the users leave is predictable with a good probability

- Generate the binary notation user Ids for all the users.
- Generate the Group Ids using the Customized Huffman technique.

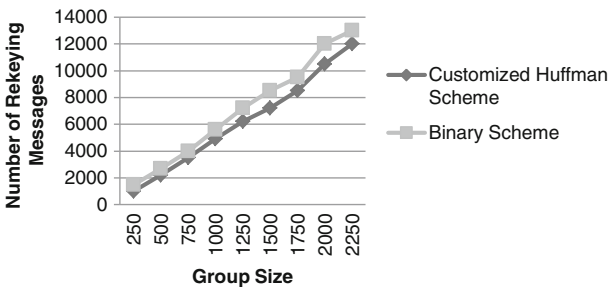

Fig. 4 Recital assessment for predictable network

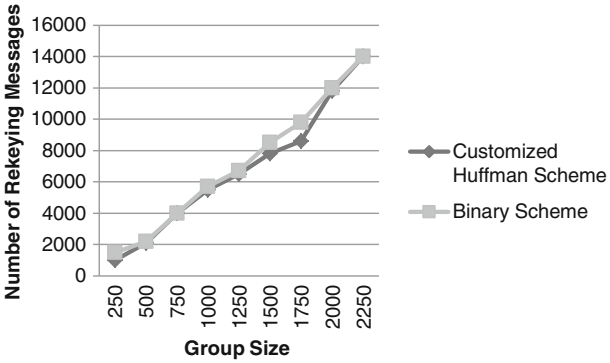

Fig. 5 Recital assessment for un-predictable network

- Generate a random number to decide who leaves next. The high probability in the generating engine should be the predicted user.
- Compute the number of re-keying operations
- The algorithm proposed in the scheme was used for computing the number of rekeying for the Customized Huffman Scheme.
- Plot a graph between these two data.

Network where the users leave is highly unpredictable

- Generate the binary notation user Ids for all the users.
- Generate the Group Ids using the modified Huffman technique.
- Generate a random number to decide who leaves next. The high probability in the generating engine should be the user not predicted.
- Compute the number of re-keying necessary for both the schemes
- The algorithm proposed in the scheme was used for computing the number of rekeying for the Customized Huffman Scheme.
- Plot a graph between these two data.

The graphical results shown in Figs. 6 and 7 are used to compare the key computation and communication time of proposed method with the existing methods. It compares the results obtained from proposed Huffman based key distribution scheme with the Lee's scheme, Tu's scheme, and Jiang et al.'s scheme.

Using the simulation the following observations are made.

1. The number of messages sent after the user leave is reduced.
2. The number of keys that has to be changed is reduced.
3. The unwanted Network traffic is reduced
4. The better our prediction of when users will leave the network, the better the recital assessment of the Network.

From the above results, it is observed that when the group size is 600, the key computation and communication time is found to be 7 μs in proposed approach, which is better in comparison with existing schemes. Moreover if the number of members who are joining and leaving increases, the computation and communication time proportionally increases. However it is less in comparison with the existing approaches.

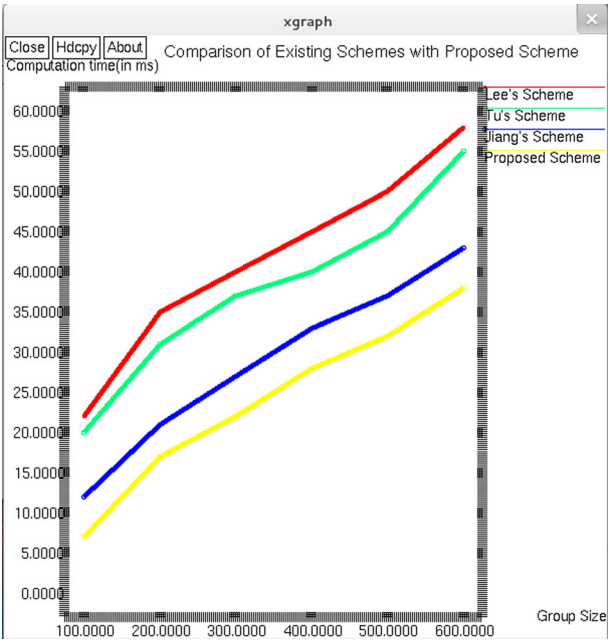

Fig. 6 Computation cost

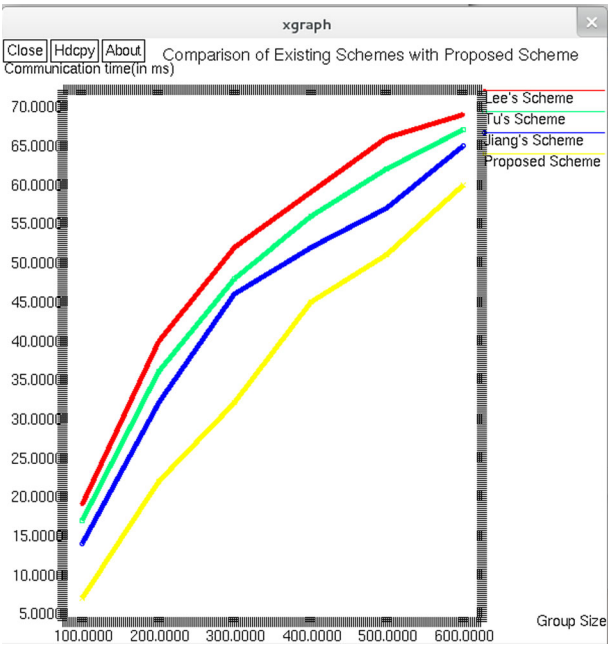

Fig. 7 Communication cost

6 Conclusion

In the proposed scheme, related works on key distribution for CAS are discussed. The proposed scheme shows the Huffman based grouping scheme for key distribution for conditional access system in DTV Multicast. By analyzing and comparing with other schemes, the proposed scheme can greatly reduce the computation and communication operations. It reduces the number of multiplication operations using fast Fourier transform and the amount of messages transferred for rekeying. CAS acquire higher efficiency and security is achieved by using extended Euclidean algorithm which is based on the difficulty of factoring large prime numbers. Moreover, the proposed scheme is more flexible in processing subscribers joining and leaving which is achieved by using Huffman based grouping scheme and is very important for service provider to dynamic manage the subscriber. Furthermore, the proposed system is well-suited for the DTV standard, which can be used for both PPC and PPV services.

References

1. Akl SG, Taylor PD (1982) Cryptographic solution to a multilevel security problem. Proc. Crypto-82, Santa Barbara, CA, August 23–25, pp 237–250
2. Antequera N, Loperz-Ramos JA (2011) Remarks and countermeasures on a cryptoanalysis of a secure multicast protocol. Proceedings of 7th International Conference on Next Generation Web Services Practices, Salamanca 2011, Salamanca (Spain) 201–205
3. Chan KC, Chan SHG (2003) Key management approaches to offer data confidentiality for secure multicast. IEEE Netw 17(5):30–39
4. Conditional Access Broadcasting Systems (1992) ITU-R Rec. 810
5. EBU Project Group B/CA (1995) Functional model of a conditional access system. EBU Tech Rev, pp. 64–77, Winter
6. Harn L, Lin HY (1990) A new cryptographic key generation scheme for multilevel data security. Comp Secur 539–546
7. Jiang T et al. (February 2004) Key distribution based on hierarchical access control for conditional access system in DTV broadcast. IEEE Trans Consum Electron 50(1)
8. Lee JW (1996) Key distribution and management for conditional access system on DBS. In: Proc. Int. Conf. Cryptology and Information Security, pp 82–86
9. Macq BM, Quisquater JJ (June 1995) Cryptology for digital TV broadcasting. Proc IEEE 83:944–957
10. Naranjo JAM, Antequera N (LG) Casado and J.A. Lopez-Ramos. A suite of algorithms for key distribution and authentication in centralized secure multicast environments. To appear in J Comp Appl Math, doi:10.1016/j.cam.2011.02.015
11. Peinado A, Ortiz A (2011) Cryptanalysis of multicast protocols with key refreshment based on the extended Euclidean Algorithm. Proceedings of CISIS 2011. Lect Notes Comput Sci 6694:177–182
12. Ray I Narasimhamurthi N A cryptographic solution to implement access control in a hierarchy and more
13. RCz S (1967) On computing the fast fourier transform. Commun AGM 10:647–654
14. Sakakibara H et al. (1994) The ID-based noninteractive group communication key sharing scheme using smart cards. In: Proc. Int. Conf. Network Protocols, pp. 91–98
15. Scott M (1990) An implementation of the fast-fourier multiplication algorithm. Technical Rep ort CA-0790, Dublin City University
16. St Denis T (2003) BigNum math implementing cryptographic multiple precision arithmetic. SYNGRESS Publishing
17. Trappe W, Song J, Poovendran R, Liu KJR (2003) Key management and distribution for secure multimedia multicast. IEEE Trans Multimed 5(4):544–557
18. Trappe W, Washington LC (2007) Introduction to cryptography with coding theory, second edn. Pearson Education, pp. 66–70
19. Tu FK, Laih CS, Tound SH On key distribution management for condition access system on Pay-TV system. In: 1998 I.E. Int. Symp. Consumer Electronics (ISCE'98), vol. 45. Taipei, Taiwan, R.O.C., pp 151–159

20. Varalakshmi R, RhymendUthariaraj V (2011) A New Secure Multicast Group Key Management Using Gray Code", Paper No: 978-1-4577-0590-8/11, IEEE-International Conference on Recent Trends in Information Technology, ICRTIT 2011,MIT. Anna University, Chennai
21. Zhu S, Jajodia S (2010) Scalable group key management for secure multicast: a taxanomy and new directions. In: Huand H, MacCallum D, Du Dz (eds) Network security. Springer, United States, 57–75

Ms. R. Varalakshmi is pursuing her Ph.D. in the area of Network Security at Ramanujan Computing Centre, Anna University, Chennai, India. She has 7 years of Teaching Experience. Her area of interests includes Network Security, Cryptography, and Information Coding Techniques. She has published various National, International Journals and Conference papers. She is the Life time member of Computer Society of India.

Dr. V. Rhymend Uthariaraj received his M.E. (Computer Science and Engineering) and Ph.D., from Anna University, Chennai, India. Currently he is a Professor and Director of Ramanujan Computing Centre, Anna University, Chennai, India. He has 27 years of Teaching Experience. He has published in various National, International Journals and Conference papers. His research interests include Network Security, Pervasive Computing, Distributed Computing, Operations Research and Computer Algorithms. He is a Life time member of Indian Society for Technical Education.