CHAPTER 3

AN IMPROVED HYBRID REKEYING MECHANISM AND ITS PERFORMANCE ANALYSIS

3.1 INTRODUCTION

A scalable and effective re-keying mechanism is required to safeguard the multicasted information and hence to enable a secure Group Communication. This part of the thesis summarizes a proposed hybrid re-keying mechanism that combines the advantages of the centralized and decentralized re-keying methods. The proposed Hybrid Re-Keying Mechanism reduces the processing load of the root server and associated networking devices using a decentralized architecture with sub-groups.

The existing hybrid architecture combines Iolus and omitted key approach of Centralized Flat Key Management (CFKM) (Waldvogel et al 1998). But the storage complexity of the algorithm is high as each member needs to keep more number of keys. Also collusion attacks are possible with the CFKM mechanism.

In this thesis an improved hybrid re-keying mechanism is proposed. In this new method, Re-keying cost is reduced by introducing separate join and leave sub trees and the storage complexity is reduced by combining the centralized approach of One-Way function Trees with the de-centralized approach of Iolus. Here collusion attacks are also prevented. Based on the
three major approaches to group key management (Centralized, Decentralized and Distributed) discussed in the chapter 2, it can be observed that,

- In centralized group key management, the single point failure very badly affects the group communication. For example, the very large group applications like Internet applications and Pay programme applications like TV pay channels and Pay web sites will not receive any programme/message if single point failure occurs and this situation continues until the single point failure is rectified. The functioning of decentralized approaches depends on particular routing protocol. In distributed key management schemes, the key management operations (communication and computations) increase exponentially as the group size increases. Thus, the currently available approaches are not cost effective in terms of computational and communication complexities.

- Group management and subsequent key management mainly depend on the dynamism of group members. Dynamic membership management is an important design feature of a multicast security protocol as members join and leave operations are very common in open networks like Internet. Most of the existing architectures do not take the behavior of the members into consideration while designing the security framework.

- Most of the architectures follow aperiodic re-keying mechanism, i.e. the keys are updated whenever there is a join/leave situation. This type of re-keying leads to two types of problems specifically, inefficiency and out-of-synchronization (Brian Zhang et al 2003).
In the above three methods each one has its own merits as well as demerits. For efficient key management if any one of the above said methods is used some sort of compromise is a must. So a hybrid re-keying mechanism Ivan (Ivan Y K Pang and H C B Chan 2003) is used. In this method they combined Iolus, a decentralized approach and CFKM which is a centralized approach.

Another novel scalable algorithm called Hybrid Re-Keying Mechanism is proposed, analyzed and implemented for establishing shared cryptographic keys in large, dynamically changing groups. This mechanism combines two different architectures namely One-Way Function Trees (OFT) and Iolus.

Among the hierarchical methods, OFT is the first to achieve an approximate halving in broadcast length. In comparison with the top-down logical key hierarchy method of Wallner (Wallner et al 1998), the OFT is a bottom-up algorithm which approximately halves the number of bits that need to be broadcast to the members in order to re key after a member is added or evicted.

The number of keys stored by group members, the number of keys broadcast to the group when new members are added or evicted, and the computational efforts of group members, are logarithmic in the number of group members.

The Iolus framework seeks to directly address the problem of scalability by completely doing away with the idea of single flat secure multicast group. Instead, Iolus substitutes the notion of a secure distribution tree. The secure distribution tree is composed of a number of smaller secure
multicast subgroups arranged in a hierarchy to create a single virtual secure multicast group.

3.2 KEYS USED IN HYBRID –REKEYING MECHANISM

The Existing Hybrid re keying mechanism (HRM) has the basic architecture of Iolus and it uses omitted key approach for re keying as shown in Figure 3.1. According to this approach each user in a group of n member has to store n-1 keys that are keys of all other members except the key having his ID. If a member has ID 2 then he will have keys k1, k3, k4, k5….. k8 in a group of 8 members. So now if this member with ID 2 leaves the new group key will be encrypted with the “omitted key” K2 so that the evicted member cannot decrypt it.

In general the keys used in HRM are explained as follows

Information Key (IK)

An Information key is employed for protecting the information. The content provider uses it to encrypt the information before sending it to all parties. Having received the protected information, a receiver can use the IK to decrypt the information. The IK is renewed whenever someone joins/departs.

Router Key (RK)

A router key is for safeguarding the communication between the content provider and all HRM-enabled routers. As the HRM-enabled routers are considered to be trustworthy, the RK can remain unchanged even if someone joins/departs.
**Group Key (GK)**

A Group key (GK) is sub-grouping based and is for protecting the communication between an HRM-enabled router and its member. The GK of a sub-group is renewed whenever someone joins/departs.

**User Keys (UK)**

A set of user keys is kept by an HRM-enabled router and the respective sub-group members. In a sub-group, the number of group members that can be supported is based on the number of UKs.

Figure 3.2 shows the different key held by the Content Provider, HRM enabled router and the user.

This scheme has two main disadvantages

- Each member has to store n-1 keys and this increases the storage complexity of the user.
- In case of bulk leave the members will be able to collude and decrypt the new group key.

An HRM-enabled router is provided for each sub-group to receive data from the root server. As far as the IP multicasting service is concerned, all parties belong to the same multicast group and has the same Information Key (IK).
**Figure 3.1 System Architecture**

**Figure 3.2 Different Keys used in HRM**
3.3 PROPOSED HYBRID–REKEYING MECHANISM (IOLUS + ONE-WAY FUNCTION TREES)

The proposed hybrid re-keying mechanism is similar to the conventional hybrid method except now in the subgroups level one-way function trees are implemented. Now there is no need to store the subgroup key since it can be calculated by each user.

![Figure 3.3 Architecture of the Proposed Hybrid Re-keying Mechanism](image)

Figure 3.3 Architecture of the Proposed Hybrid Re-keying Mechanism

3.3.1 Member Join and Leave

In case of member join the new IK is encrypted with old IK and sent to HRM enabled router and members. The HRM enabled router renews the keys along the path to the root of the subgroup and sends the renewed blinded node secrets to siblings. It then sends a unicast message which
consists of node secret, blinded node secret of its siblings and the new IK to the newly joined member.

In case of ‘member leave’ the node secret of sibling nodes is renewed and this changes the node secrets from the leaving member’s node to the root and this automatically changes the GK. The renewed keys are sent to the sibling nodes. The IK is renewed by root and sent to the HRM enabled router using the router key (RK). The HRM enabled router sends the new IK encrypted with the new GK. The leave operation is performed, immediately followed by the join operation during re-keying, which improves the efficiency of the re-keying process.

3.4 IMPLEMENTATION OF JOIN AND LEAVE IN HRM USING ICKAP

In secure group communications, the time cost associated with the key updates in the event of member join and departure is an important aspect of quality of service, especially in large groups with highly dynamic membership. A new method has been proposed in the same hybrid re-keying mechanism architecture with different re-keying technique. To achieve better time efficiency, a join and leave tree management frame work has been introduced. A special key tree topology with join and leave sub trees is introduced to handle key updates for dynamic membership. Then optimization techniques are employed to determine the capacities of join and leave sub trees for achieving the best time efficiency and algorithms are designed (Interval Based Cost effective Key Agreement Protocol) to dynamically update the join and leave trees. It has been shown that on average the asymptotic time cost for each member join/departure event is reduced to $O(\log(\log n))$ from the previous cost of $O(\log n)$. Where $n$ is the group size. The experimental results based on the simulated user activities demonstrate
that the proposed re-keying mechanism can significantly improve the time efficiency while maintaining the low communication and computation cost of tree based contributory key agreement. Due to reduced re-keying cost the number of users in a group can be increased which improves the scalability of the system.

![Diagram of Hybrid Key Management System](image)

**Figure 3.4  Implementation of Join and Leave Functions in HRM using ICKAP**

In this part of the thesis, a new hybrid scheme combining the one way function trees and Iolus is presented. Unlike the previously proposed hybrid scheme combining CFKM with Iolus this new scheme has less storage requirements and also avoids collusion attacks. The communication, computation and storage requirements scale logarithmically with group size. This hybrid scheme offers scalability and with low broadcast size to manage
the demanding key establishment requirements of secure applications for large dynamic groups.

3.4.1 Comparative Analysis

Comparison of the complexities involved in the existing and the proposed hybrid architecture is tabulated below and all the complexity analyses are discussed below.

### Table 3.1 Analysis of the Existing and Proposed Hybrid

<table>
<thead>
<tr>
<th>Complexities Involved</th>
<th>Existing Hybrid Method (IOLUS+ CFKM)</th>
<th>Proposed Hybrid Method 1 (IOLUS + OFT)</th>
<th>Proposed Hybrid Method 2 (IOLUS+ICKAP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key complexity</td>
<td>m (n + 1) + 2</td>
<td>m (n + 1) + 2</td>
<td>m (n + 1) + 2</td>
</tr>
<tr>
<td>Storage complexity</td>
<td>m (n^2 + 5n + 6) + 5</td>
<td>((log(n+4)n+3)m+2)</td>
<td>log((log(n+4)n+3)m+2)</td>
</tr>
<tr>
<td>Communication Complexity</td>
<td>join 2(m+1)</td>
<td>log(2(m+1))</td>
<td>log(log(2(m+1)))</td>
</tr>
<tr>
<td></td>
<td>Depart 2(m+1)</td>
<td>log(2(m+1))</td>
<td>log(log(2(m+1)))</td>
</tr>
<tr>
<td>Computation Complexity</td>
<td>Join m+5</td>
<td>log(m+5)</td>
<td>log(log(m+5))</td>
</tr>
<tr>
<td></td>
<td>Depart 2m+3</td>
<td>log(2m+3)</td>
<td>log(log(2m+3))</td>
</tr>
</tbody>
</table>

3.4.2 Complexities Involved in Existing Hybrid Method 1 (IOLUS+CFKM)

Notations

n - number of members in each subgroup
m - number of subgroups
**Key Complexity**

There are $n \times m$ user keys where $n \times m$ is the total number of members.

There are $m$ group keys.

One Router key

One Information key

**Total number of keys** $= m \,(n+1) + 2$

**Storage Complexity**

The storage complexity of this algorithm can be calculated as:

Total storage $=$ No of keys stored + hashing function + key count + hash count

Keys held by the Root $= 1\,IK + 1\,RK$

$= 2$

Keys held by the HRM Router $= (IK + RK + GK + n \, Uks) \, m$

$= (1 + 1 + 1 + n) \, m$

$= (3 + n) \, m$

Keys held by the Members $= (IK + GK + (n-1) \, UK) \, nm$

$= (1 + 1 + (n-1)) \, nm$

$= (n+1) \, nm$

Total key storage $= 2 + (n+3) \, m + (n+1) \, nm$

$= 2 + mn + 3m + n^2m + nm$

$= mn^2 + 2mn + 3m + 2$

$= m \,(n^2 + 2n + 3) + 2$

Total storage $= m(n^2 + 2n+3) + 3( \, nm + m +1) + 2$

$= m \,(n^2 + 2n+3 + 3n+3) + 5$

$= m \,(n^2 + 5n+6) + 5$

Thus the storage complexity increases in the order of $o \,(n^2)$. 
Communication Complexity

One Member join

IK-Renew : ‘m’ multicast message from root to all members and HRM routers.(2m)
GK-Renew : One multicast within subgroups by HRM to all members.(1)
Unicast : One unicast message to newly added member with new IK, new GK and UKs except his own UK from HRM router. (1)

Thus the communication complexity during member join is (2m+2)

One Member depart

GK_Renew : ‘m’ multicast from HRM router to members and one unicast to root(1). (m+1)
IK_Renew : 1 multicast from root to HRM router and m multicast from HRM router to members of each subgroup.(m+1)

Thus the communication complexity during member leave is 2m+2 = 2(m+1).

Computation Complexity

One Member join

IK-Renew : 1 encryption using old IK and m decryptions(m+1).
GK-Renew : 1 encryption using old GK and 1 decryption (1+1)
Unicast : 1 encryption using private key and 1 decryption (1+1).
Total= (m+5)
One Member Depart

GK-Renew :  
1st encryption using old GK.  
2nd encryption using Omitted key.

IK-Renew :  
1st encryption using Router key and ‘m’ encryptions 
‘m’ decryptions for m sub groups.

Total = (2m+3)

3.4.3 Complexities Involved in the Poposed Approach Iolus with OFT (HRM)

Notations

n-number of members in each subgroup. 
m-number of subgroup. 

Key Complexity

There are n x m user keys where, n x m is the total number of members. 
There are m group keys. 
One Router key. 
One Information key 
Total number of keys = m (n+1) +2

Storage Complexity

Total storage = keys storage + storing hashing function

Keys held by the Root = 2
Keys held by the HRM router = (1 RK+1 IK+ n Uks) * number of sub-groups
= (2+n)m

Keys held by the Users = (1+logn+1) mn

where,

log n - blinded node secrets of siblings.
1 node secret +1 IK
Hashing function storage = (nm+m)
= m(n+1)
Total storage = 2 + (2+n)m +(2+logn)mn+ m(n+1)
= ((log n+4) n+3)m+2

In this case the storage complexity reduced to the order of log (n).

Communication Complexity

One member join

IK-Renew: One multicast message from the root to all the members and ‘m’ HRM routers (m+1).

Unicast: One unicast message to newly added member containing IK, node secret and blinded node secret of siblings from HRM router. ‘m’ multicast message containing blinded node secrets of nodes from the new node to the root(m+1). Hence the total communication complexity for member join is (2m+2). As the One way Function Tree (OFT) is introduced the complexity reduces to O(logn)

Thus the communication complexity during member join is log (2(m+1)).
**One Member depart**

**Unicast**: Message to the departed member’s sibling giving him a new node Secret (1)

**Multicast**: Message to the other members to inform them about the new blinded node secrets along the path from node to root (m)

**IK_Renew**: 1 multicast from the root to HRM router using Router key and m multicast from the HRM router to members of each subgroup using new Group key(m+1).

Communication complexity = \( \log (2(m+1)) \)

**Computation Complexity**

**One Member join**

**IK-Renew**: 1 encryption using old IK and m decryptions(m+1).

**GK-Renew**: 1 encryption using old GK and 1 decryption (1+1)

**Unicast**: 1 encryption using private key and 1 decryption (1+1).

(m+5) As the One way Function Tree (OFT) is introduced the complexity reduces to \( O(\log n) \). Hence the computational complexity becomes \( \log (m+5) \).

**One Member Depart**

**GK-Renew**: 1\(^{st}\) encryption using old GK.

2\(^{nd}\) encryption using Omitted key.
IK-Renew : 1 encryption using Router key and ‘m’ encryptions
‘m’ decryptions for m sub groups.

The total computational complexity for leave becomes \((2m+3)\), as the One way Function Tree (OFT) is introduced the complexity reduces to \(O(\log n)\). Hence the computational complexity becomes \(\log(2m+3)\).

### 3.4.4 Enhanced Hybrid Rekeying Mechanism Using ICKAP
(proposed and enhanced re-keying)

**Notations**

- \(n\) - number of members in each subgroup.
- \(m\) - number of subgroup.

**Key Complexity**

There are \(n \times m\) user keys where, \(n \times m\) is the total number of members.
There are \(m\) group keys.
One Router key.
One Information key
Total number of keys = \(m(n+1)+2\)

**Storage Complexity**

Total storage = keys storage + storing hashing function
Keys held by the Root = 2
Keys held by the HRM router = \((1RK+1 IK+nUks)\)*number of sub-groups
\(= (2+n)m\)
Keys held by the Users = \((1+\log(\log n+1))mn\)
where,

\[
\text{Log}(\log n) - \text{blinded node secrets of siblings.}
\]

1 node secret +1 IK

Hashing function storage = \((nm+m)\)
\[
= m(n+1)
\]

Total storage = \(2+(2+n)m+(2+\log(\log n))nm+m(n+1)\)
\[
= \log(((\log n+4)n+3)m+2)
\]

In this case the storage complexity reduced to the order of \(\log(\log (n))\).

**Communication Complexity**

**One member join**

IK-Renew : One multicast message from the root to all the members and ‘m’ HRM routers (m+1).

Unicast : One unicast message to newly added member containing IK, node secret and blinded node secret of siblings from HRM router. ‘m’ multicast message containing blinded node secrets of nodes from the new node to the root(m+1). Hence the total communication complexity for member join is (2m+2). As the ICKAP protocol implemented with join and leave trees the re-keying complexity reduces to \(O(\log(\log n))\)

Thus the communication complexity during member join is \(\log (\log (2(m+1)))\).
One Member depart

**Unicast**: Message to the departed member’s sibling giving him a new node Secret (1)

**Multicast**: Message to the other members to inform them about the new blinded node secrets along the path from node to root (m)

**IK_Renew**: 1 multicast from the root to HRM router using Router key and m multicast from the HRM router to members of each subgroup using new Group key (m+1). As the ICKAP protocol implemented with join and leave trees the re-keying complexity reduces to $O(\log(\log n))$.

Thus the Communication complexity for member leave = $\log(\log (2(m+1)))$

**Computation Complexity**

**One Member join**

**IK-Renew**: 1 encryption using old IK and m decryptions (m+1).

**GK-Renew**: 1 encryption using old GK and 1 decryption (1+1)

**Unicast**: 1 encryption using private key and 1 decryption (1+1).

(m+5) as the ICKAP protocol implemented with join and leave trees the re-keying complexity reduces to $O(\log(\log n))$, thus the Computation complexity for member join = $\log(\log (m+5))$. 
One Member Depart

GK-Renew : 1\textsuperscript{st} encryption using old GK.
2\textsuperscript{nd} encryption using Omitted key.

IK-Renew : encryption using Router key and ‘m’ encryptions
‘m’ decryptions for m sub groups.

The total computational complexity for leave becomes(2m+3), As the ICKAP protocol implemented with join and leave trees the re- keying complexity reduces to O(log(logn)).

Thus the Computation complexity for member leave = log(log (2m+3)).

Secure multicast communication is a challenging problem in today’s networks. The proposed model is based on the hybrid re-keying approach. For key distribution among the group members, a novel key tree called the join and leave tree is defined. This virtual key tree is compared for its performance with the existing hybrid re-keying model. The experimental analysis show that this model minimizes the overheads like storage, communication efficiency, encryption cost, the number of re-key messages sent due to join/leave and the total processing cost compared to other methods. Therefore, this is the most suitable security framework for IP models that satisfy the requirements of today’s applications. In this approach, when the group size is increased, the cost associated with the key updates also increased. To reduce the time cost to an optimum level an Interval based cost effective key agreement (ICKAP) approach is proposed and its detailed discussion is follows in the next section.