Chapter V

A MODEL FOR SEARCHING IN AN ENCRYPTED DATABASE ENCRYPTED USING TDMRC CODE

5.1 Introduction

5.2 Feature of Proposed Technique

5.3 System Model

5.4 Technical Aspects

5.5 Technique for searching TDMRC code Based Encrypted Database

5.6 Performance and Improvement

5.7 Application of Proposed Model on Real Time System

5.7.1 Security and Correctness of the Proposed Algorithm
5.1 Introduction

With the improvement in technology and network connectivity, a mandatory requirement for many organizations is to have a multiuser environment in the database which is connected via local server or through Internet. In recent years, storing the data in an outsourced database is also seen with most of the users opting for it without hesitation. For security reasons encrypting of the database is often done so that sensitive or critical data is secure in a distrusted database. A database system designed to handle workload whose state is constantly changing is called a real-time database system. This is different from traditional database containing persistent data most of them not affected by time. [BUA 2005] Real time processing means that the transaction is processed with a very high speed and the results gets back to be acted upon right away [KVA 2006]. Real time database are useful for stock market, banking, reservation system, scientific data analysis etc.

With the already published proof [VP 2003] TDMRC code when implemented in FTHRT (Fault Tolerant Hard Real Time System) will act as a sleeve over the communication channel protecting it from eaves dropping. The main disadvantage of the system is that searching on the encrypted database is not made possible. Once encrypted by TDMRC code the database system becomes unapproachable for an intruder as well as the genuine user. When the user wants to get back the data related to a specific keyword for example “better” there are two extreme ways to do that. First, by decrypting the entire
encrypted database now better known as 'cipher text' and then searching for the keyword. If the database is located at a different server then there should be an additional cost incurred for encrypted database now better known as 'cipher text' and then searching for the keyword. If the database is located at a different server then there should be an additional cost incurred for downloading the entire database too. Second method is to identify the possible area where the keyword “better” lies and then decrypting only that area. The first method is not a good solution to the problem as it requires much transfer between the server and the client if both of them do not happen to be the same. Second method is a better solution for the problem and in this research work keyword based search is considered. Searchable encryption allows user to perform keyword based search on encrypted database just as in traditional database transaction. [CGK 2006, CMM 2005, EJG 2003, DXS 2000] These were based on single user setting. Multiuser searches were allowed with extending the capability of single user later [CGK 2006], where only one user is allowed at a time to update the database where as others could only view it. Here the idea of sharing the secret key among many users was done.

In this experimental work searchable encryption in multi-user environment is considered. The features of the algorithm developed for the system is given in the following section followed by its model. The algorithm’s performance and application are discussed towards the end of the chapter.

5.2 Features of the proposed technique:

- This system has an user admin that allows a group of users each possessing a distinct username (given to the user at the time of
registration to the system—discussed in chapter IV

4) and password (of their choice) to insert their data records to the database with each record they insert having distinct keyword. Which means a single user is not lowed to use same keyword for more than one record, whereas a different user can use the keyword (used by another user) for his/her record.

- In the database the data record inserted will be saved in an encrypted format encrypted using TDMRC code (discussed in chapter 3)

- This system allows dynamic user registration, since a user joining does not affect other user's performance. Any user is allowed to access only those records which they have permission to.

- This system does not allow a dishonest user or the database server to generate valid queries on behalf of another user even if the user's secret key (username is compromised).

This system was developed based on encrypted database encrypted using TDMRC code where different users are allowed to enter their data records along with the keyword which has restricted access. There is no sharing of the data done and only the originator of the data record is allowed to write to and search the data records without sharing their secret keys.

5.3 System Model

Consider a database system \( \{D, UA, S, U\} \), where \( D \) is a database; \( UA \) is
the user admin of the data owner organization that is responsible for managing
the users, e.g., user registration and user logout session; S is the database server
providing the search service; U is a group of users. The database D consists of
m records \{r_1, r_2, ..., r_m\} of multiple attributes. One of the attributes is the
keyword used for search. The domain of the keyword attribute is denoted by K.
The keyword of \(r_i\) is denoted by \(r_i\):k. The server S does not provides D directly
instead it is made available by encrypting it by TDMRC code.

This can be denoted by \(D'\), where \(r'_i = \langle E(r_i) \rangle\), where \(E(r_i)\) is an
encryption of \(r_i\).

Here the keyword to a particular record is made to be given by the user.
The advantage of this method is that user can uniquely identify each record
with the help of keyword and does not have to depend on automatically
generated keyword which may not make any sense. For example if the user is
trying to save a set of mails then he can save it keeping the sender name as the
keyword or date or he can even chose a word in the mail as the keyword to the
record. Here the assumption is that each user will have a specific keyword
assigned to only one record and each record will hence have a unique keyword.

Here let \(E_0 = \{E(r_1), ..., E(r_m)\}\) and \(K_{wd} = \{(r_1:k), ..., (r_m:k)\}\).

After the authentication done by the server S, an authorized user \(u \in U\),
is allowed to insert data records to \(D'\) and to search data records inserted by \(u\)
and by others based on her access rights on those records. Whenever the user \(u\)
request the server $S$ for a specific keyword using the query $Q_u(k)$ to denote a query from user $u$ on keyword $k$, the server $S$ is expected to return $p_x=\{<E(r)>, r \in D, r.k=k\}$.

User admin, $UA$ keeps the track of the user $u$ in the database. The user $u$ gets registered in the database with $UA$ providing the unique username to the user. The username calculating algorithm has been explained in chapter 4. $UA$ has power to rescind a user's privilege of searching the database. Therefore, the user set $U$ is divided into an authorized user set $AU$ and a rescind user set $RU$. Only users in set $AU$ are allowed to successfully search and write to the database and those not in $RU$. Here $UA$ is considered to be having the power of system administrator of the database server but is not the one. Apart from registering the user $UA$ also allows user to enter the data record and keyword for each record and that the keyword has to be unique among the different record-keyword pair the user has entered. Here the study has been done with the assumption that $UA$ is semi-trusted admin.

The multi user encrypted database system which is considered here consists of the following algorithms:

- **TDMRC** $(n, S_m)$. An encryption algorithm executed by $UA$ to encrypt the data records written by the user $u$. Here $n$ is the Poly alphabetic Coefficient value and $S_m$ is the $n$ number of sub keys required by the algorithm. The Master Key is read from the system
clock at the time of encryption. The output of the algorithm is the data record in an encrypted format for a secure storing of the data record.

- Username generator (DTn). This algorithm generates unique username with the date time DTn passed as input to it. The output of this algorithm is the unique username which can be assigned to user (explained in chapter 4).

- Register (P_u). This algorithm is executed by UA to register the user u to the system. The algorithm takes as input the username generated by the Username generator and the password given by the user u.

- Keyword Gen (k) For every record r_i entered to the database server the user u requests for keyword k. The keyword if it is unique to the user kept domain of keywords the algorithm outputs the permission to use the keyword. This algorithm is meant to keep unique keyword for each data record the user enters

- Write (r) This is run between user u and the server S to write an encrypted record of r_i to D'. This is done together with the Keyword Gen algorithm.

- Search (Qu(k),D') This algorithm is run by the server to search D' on request of an authorized user AU' for records with keyword k.
• Rescind \((u)\) Run by \(UA\) to cancel a user from doing any kind of updating or viewing of the record for the present moment. As a result of this \(u\) no longer has ability to view or search the database.

• Link \((k)\) Link the keyword to the relevant data record. Each data record \(r_i\) has a keyword associated with it denoted by \(k_w(r_i)\)

The multi-user encrypted database system proposed can be described as working correctly if following features happen to be true.

➢ If an authorized user \(u \in AU\) gets the correct record with the query he generates.

The security requirements that have to be followed are:

➢ If an unauthorized user or a rescind user \(u \in RU\) is denied the search of any record he is trying for.

➢ Privacy of query is another common security requirement for all searchable encryption schemes, which is the amount of information leakage to the server regarding the user queries. The privacy of the query has been discussed by researchers in their papers [CGK 2006, BCG 1997]. It is to be noted that for searchable encryption server always observes the database access patterns but still should not learn about the keyword. Apart from the
information obtained from the observation the server should not be
given any access to any other information.

➢ In the model proposed for a record \( r_i \), \( k_{w_i}(r_i) \) denotes the
keyword to the specified record. Whenever the authorized user \( AU \)
enters a paragraph /record to the server for storing the record
along with the keyword is encrypted by TDMRC code and
reaches the database in an encrypted format. So having the
keyword in an encrypted format keeps it very secure in the server as
well as from curious system administrator. Here we consider the
system administrator as a trusted but curious one to know about
information. From the query privacy what is meant is that the server
should be given only that piece of information that we pass to the
intruder (if there is any) also. Which in the other way can be quoted that
it is the record secrecy which we really aim at and the encrypted
database \( D' \) should not reveal information about the original database.

➢ In the proposed system the user is allowed to issue a query only when
the user becomes authorized. The user \( u \) is said to be authorized if the
user is able to successfully complete the username and password
checking state of the software. The authorized user \( u \) can now issue a
query on the records with the help of keyword. If any
unauthorized access happens (chances are very rare as it should
compromise on the user \( u \) secret keys) the unauthorized user will
only be able to access the encrypted keywords and encrypted data records which are encrypted by TDMRC code. This in turn will be of no use to the user as cipher text with TDMRC code is impossible to attack [VP2003]. Thus neither another user nor the server can generate exact query on behalf of authorized user $u$. With this method the secrecy of the keyword is maintained and the keyword is only known to its originator and not to anybody else as it is in encrypted format.

- Option to rescind the user $u$ at any point of time is an added advantage to the system being proposed. Here the behavior of the user is checked after entering the system. If more than a selected $n$ attempt to search the database ($n$ being less than 4) by a user $u$ fails then the specific user is cancelled or is rescind by the user admin $UA$. User cancellation is a mandatory part of multiuser application. Whenever there is an inappropriate behavior or the believed legitimate user $u$ fails at the privacy check then the user is cancelled immediately and is not allowed to search or view the database from then onwards. The user has to contact the $UA$ to change the password to get access to the database.

- An intruder has an advantage if he is able to access the system pretending to be a legitimate user and then inserting a data record $j$ into the database along with a keyword $k_{wd}(j)$. He will
have only the advantage of inserting an unwanted record and not searching through the database and getting data record $r_i$ without knowing the exact keyword $k_{wd}(r_i)$.

5.4 The Technical Aspects

As a part of the research software was developed during the progress of this work, which had the logic of TDMRC code and which could encrypt the data with any number of sub keys and any value taken as Poly Alphabetic coefficient. The Master key is taken as the clock time at the time of encryption and the sub keys, Poly Alphabetic coefficient fed through the keyboard. Using the software decrypting could also be done with a known key. The database used for the model was SQL Server.

5.5 Technique for Searching in TDMRC code based encrypted database

1) Obtain the username and password from the user $u$. If the username and the password given by $u$ succeed the security check then continue.

2) If the user wants to enter a data record then proceed to step 3 else if the user wants to search and retrieve the data record then proceed to step 13.

3) Get the data record $r_i$ given by the user
4) Let the user \( u \) (originator of the data record \( r \)) decide unique keyword for the data record \( r \), denoted as \( k_{ud}(r) \). Poly alphabetic coefficient \( P \) and corresponding number of sub key (eg, subkey1, subkey 2… sub key n). Apart from the keyword all other values are stored as a header file to the data record.

5) Get the keyword from the user.

6) Find the ASCII value of each character of the keyword.

7) Add ASCII value of first character with ASCII value of second character append with it the sum of ASCII value of third character and ASCII value of fourth character append with it the sum of ASCII value of fifth character and ASCII value of sixth character – this process is to continue till the last character.

8) Convert the above result to number

9) Obtain the left most 4 digits, denoted by keynum.

10) Add with it the subkey1 to get original first sub key to be passed to TDMRC code

11) Add with the result obtained in step 10 to keynum to obtain the original second sub key to be passed to TDMRC algorithm. Depending on Poly Alphabetic coefficient chosen the number of sub key may vary. This process continues till the \( P^{th} \) sub key is obtained, where \( P \) is the Poly Alphabetic Coefficient.
12) After getting the required number of original sub keys depending on the Poly Alphabetic coefficient PAC, and Master key from the system clock, encrypt the data record associated with the keyword. Append to the data record sub key 1, date time (at the time of encryption), P

13) Get the keyword from the user.

14) Find the ASCII value of each character of the keyword.

15) Add ASCII value of first character with ASCII value of second character append with it the sum of ASCII value of third character and ASCII value of fourth character append with it the sum of ASCII value of fifth character and ASCII value of sixth character – this process is to continue till the last character.

16) Convert the above result to number

17) Obtain the left most 4 digits

18) Add with it the subkey 1 to get original first sub key to be passed to TDMR C code

19) Add with the result obtained in step 18 to keynum to obtain the original second sub key to be passed to TDMRC algorithm. Depending on Poly Alphabetic coefficient chosen the number of sub key may vary. This process continues till the $P^n$ sub key is obtained, where $P$ is the Poly Alphabetic Coefficient.

20) After getting the required number of original sub keys along with the Poly Alphabetic coefficient $P$, which is stored as the header to data record in TDMRC code encrypted format and stored Master
key (when encrypted) decrypt the data record associated with the keyword.

The central idea of this method is like the database being divided into different logical section based on username and keyword. The main advantage of this method is that with this scheme each user can be assigned as much space he requires in database and keyword search will only decrypt the paragraph/data record that corresponds to the keyword. The security is improved with the number of key needed by TDMRC code reduced and the functionality is still intact. With this technique only one sub key needs to be kept in the server along with the Poly Alphabetic Coefficient PAC and the Master Key. With this improvement even if the intruder tries to access the system and the record he will not be able to succeed since he has only one sub key with him and no clue about the key word.

TDMRC code modification suggested here with the search is that it is completely dependent on keyword. After a series of calculation done with the ASCII value of each character in keyword the final answer obtained is added to the sub key(stored as a part of encryption) to obtain the actual sub key to be passed on to TDMRC decryption stage. With this the security of the key management of TDMRC code is greatly improved as compared to the earlier case where the master key as well as the sub key along with the Poly Alphabetic Coefficient is stored as separate file and hence the key management was very poor.

Without the user u knowing the exact keyword the decryption of the data record will never happen. With a different keyword
given only a set of ASCII characters will be returned back which will make no sense at all.

The following example shows the output of decryption using different keyword

Plain Text: “Yesterday is but a dream; tomorrow is but a vision, but today well lived makes every yesterday a dream of happiness, and every tomorrow a vision of hope. Look well, therefore to this day.”

RTC TIME 10:23:34.67
MASTER KEY 10233467
PAC 3
SUB KEY 1 2345
KEYWORD Dream
ASCII value of each character of keyword 13 114 101 97 109
Calculation done with keyword 13+114 || 101+97 || 109 = 127 || 198 || 109
= 127198109

Take the leftmost 4 digit =1271
Original sub key1 =1271+2345=3616
Since PAC=3, the system generates two more sub keys Sub Key2, Sub Key3
Sub Key2 = SubKey1+1271=4887
Sub Key3 = SubKey1+1271=6158
Therefore with PAC=3, Sub keys – 3616, 4887, 6158, Master Key- 10233467
Cipher Text - 9Xiot_p\'z it\'uw_^Ue\'glt(mqnr\_v't\'uqite(o+\'vt tqR_y uemcleqehm\hsu@V\_pwz@s*\_Ud'S\qiZmm\VZqpele toorldu@V\_pwwlmopUpx_vfs\Wlnm\V1q_/Lqlgwdlm,*c dqi\ps_tq_tcen hZxUB

If the keyword used is changed to dreams = 100 114 101 97 109 115

Calculation done on keyword to obtain the sub keys as – 4486, 6627, 8768

Therefore with PAC=3, Sub keys – 4486, 6627, 8768, Master Key- 10233467

Cipher Text obtained – \_eDtu\'s\_e\)%8q\'utcgs\']24 tm"mlsnwjdpul\vjq8mm. dpttm\_liwDAAl8\'_2cVDq\'vDs%iDqu\'sgcic\_1\'c2 nehhjojmDqt.JmgDv\'1tn\"ms1mywjqjmmehmoD vBmmkwDl(A)uhDsDemsDtnu(jq1tn\"ms1\_jiv>my jwjqjmmehmoDvBmmkwDl(A)uhDsDemsDtnu(j q\_jiv>

If the keyword used is changed to dReam = 100 82 101 97 109

Calculation done on keyword to obtain the sub keys as – 4166, 5987, 7808

Therefore with PAC=3, Sub keys – 4166, 5987, 7808, Master Key10233467

Cipher Text obtained – DS\iqdpcZ0hdoh\tshuh.m\Zi6\"rlQnmQ,[ho":rshZh sdlg\"ts[qi.Zx[u\kh"k'b\"iZj\"s\db\"mx\"x\or\m_Zy[ Z\_m\ul[ia\gjEdm\os,hZkchdb\ny[rilQpplv\"Zh\'sh ik[l][YQo\FhVlia[u\kc\"rYdndaQm\hs[lqYhs[.Zx.>

99
We can note an avalanche effect when the subkeys are changing. Hence from
the security aspect we can say that without the PAC, Sub key 1, Master Key and
the keyword the intruder will never be able to get back the original plain text.

5.6 Performance and Improvement

For user registration For user registration there is no need for checking
(as explained in chapter 4) for availability of the username. Hence the
complexity here is $O(1)$.

For query process For a query process, the main computation at the
server side includes a set of arithmetic operations involving addition alone
done on the keyword, decryption of the data record associated with the
keyword the computation cost is linear to the size of keyword set.

For storing of keyword For each data record the algorithm proposes to
store a corresponding keyword this storage requires a space complexity of
$O(m)$

Since all the existing scheme with single user and multi user require
$O(m)$ server computation the efficiency of the proposed scheme does not
downgrade due to key management and multi user capability.

The system is capable to work in a multi user environment but sharing of
the data records are not considered/allowed. The study was limited to the
assumption that the data records were at most important to the user and none
other than the user should have access to them. The software can however be
extended with a secret key sharing among the users to get such facilities.

5.7 Application of the Proposed Model on Real Time System

One of the important applications considered here is to encrypt the data and send it across through an email to a trusted party where with the same software the trusted party can decrypt it.

This application was done and tested with Pentium machines (with the proposed software) at both ends called as station.

The data to be send through the mail was encrypted using the software and send across to the receiver station with a header containing information about Poly Alphabetic Coefficient, Sub Key1, Master Key in a TDMRC code based encrypted format. Details of decrypting the header and the keyword are already passed to the receiver station in advance.

Encrypted cipher text transmitted at different time is given in following example. It is decrypted to obtain the plaintext towards the end of the example.

Time: 07:56:15

806/9→7L1MA98Leb<eYxpwBrgafcnn_.\}<_<nfarsobp]ne`lvjb^a<~
kod b9nu-%Fdo byboy Wj1ii[cfb fk0dv~aiaZ nedpd _< ko[linpcfq`
\9%ioqtqk;?

Time: 08:05:10
Data received at the receiving station is given below. From the example it was observed that the encrypted data received at the receiving station when decrypted gave the same plain text.
Decrypted mail:

Keyword: growth

“The pain you feel today is the strength you feel tomorrow. For every challenge encountered there is opportunity for growth.”

Keyword: grace

“Happiness cannot be traveled to, owned, earned, or worn. It is the spiritual experience of living every minute with love, grace & gratitude.”

Keyword: Responsibility

Accept responsibility for your life. Know that it is you who will get you where you want to go, no one else.

5.7.1 Security and Correctness of the Proposed Algorithm

Security check was also done with a third party (taken as an intruder) obtaining the email and trying to get back the plaintext. Since the details of PAC, Sub Key1, Master Key were in encrypted format and looked similar to the cipher text; picking out the header file from the source file itself was found to be a difficult task.

Plaintext: Once there was a little pink Rosebud, and she lived down in a little dark house under the ground. One day she was sitting there, all by he

The encrypted paragraph obtained by the third party

6:.48'AL5-.79?on"SFos(XpaqUftqFguo`mutjqqCC`Vrx`lUcUemx5lub3qj
The header file was also separated and the security check was done with the cryptanalyst given the ciphertext alone. Still it was of no use since the value of sub Keys depends on keyword which was unknown. Hence the correctness of the algorithm was observed to be true.

The encrypted paragraph after the exclusion of the header file

Applying the proposed model on Real Time database system was tried with bulk data created in an Excel sheet which was updated very often with no specific time period between two updates. Keeping Excel Sheet as the source file and word as the destination file the required encrypted cells of Excel sheet was copied to word document as a link file. TDMRC encryption was done on the entire file at a time keeping file name as the keyword and the Master key read from the system clock, with the sub key kept as fixed. Whenever there was an update in Excel the corresponding document was updated automatically.

Application was tried again with each user’s transaction being encrypted
based on the username, password, keyword where password and keyword were provided by user.

The bulk data considered was Bank statement from ABC Bank (Name changed for confidentiality)

The setup was later extended to two systems in which system1 had the Excel sheet which was very often updated. Every time it gets updated the data was sent for encrypting using TDMRC code. Here the Master Key used was the time at which the file was retrieved for encryption and fixed Sub Key1. Initially the Filename itself was used as keyword. In System 2 a job was scheduled to retrieve data from system1 every one hour. With it working, the same application was tried again with each user's transaction being encrypted based on the username, password, keyword where password and keyword were provided by user.

Plate 10 shows the encrypted bulk data in an instance and Plate 11 shows search on bulk data.
PLATE #10 : Encrypted Bulk Data

PLATE #11 : Search on Bulk Data

Username : 13715807291q3942
Password : 123
Keyword : 020512

Obtained Plaintext :

ABC Bank Limited

<table>
<thead>
<tr>
<th>Date</th>
<th>Chq No</th>
<th>Clearing Date</th>
<th>Debits/Withdrawals</th>
<th>Credits/Deposits</th>
<th>Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>02/05/12</td>
<td>000000829612</td>
<td>02/05/12</td>
<td>30,000.00</td>
<td></td>
<td>-2,489,274.54</td>
</tr>
</tbody>
</table>

Username : 13715807291q5470
Password : 1234
Keyword : bknst

Obtained Plaintext :

ABC Bank Limited

<table>
<thead>
<tr>
<th>Date</th>
<th>Chq No</th>
<th>Clearing Date</th>
<th>Debits/Withdrawals</th>
<th>Credits/Deposits</th>
<th>Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/11/12</td>
<td>31/10/12</td>
<td>31/10/12</td>
<td></td>
<td></td>
<td>33,186.00</td>
</tr>
</tbody>
</table>
Username : 13715807291q5470
Password : 1234
Keyword : Bnkst
Plaintext : No Response
Keyword : BNkst
Plaintext : No Response

Keyword : BN123
Plaintext : No Response
EXIT

Username : 13715807291q5470
Password : 1234
Invalid User

Username : 13715807291q5470
Password : 12345
Keyword : bkst+MICR

PLATE # 11 : Search on Bulk Data
Password : 123
Keyword : bnkst
Invalid User

Username : 1371580729I9999
Password : 1234
Invalid User

Username : 1371580729I5470
Password : 1234
Invalid User

PLATE # 11 : Search on Bulk Data

From Plate #10 and Plate #11 it can be seen that when username, password and the keyword are given by the user correctly then the system decrypts the required field alone and gives it back to the user. This can be found from the first, second and last three instances shown in Plate #11.

The system does not allow the user to provide the wrong keyword for more than three times in a particular transaction. System quits automatically
when a wrong keyword was tried for the third time. Later when the same user tried to login the user was identified as invalid user.

In the last instance of Plate #11 it is seen that the user had to change his/her password with the permission of the administrator to login subsequently to retrieve the record.

The proposed system can be applied to any Real Time system which requires high security and high communication speed because of TDMRC codes’ ease of calculation will be an added advantage when a high speed communication is required along with highly secure encryption technique. More over the authentication and retrieval of data records are done with the searching technique proposed.

**Conclusion:**

This chapter dealt with the technique developed to search in an encrypted database encrypted using TDMRC code. The details of the algorithm, its complexity and its working is explained in this chapter. Application of this algorithm is also specified towards the end of the chapter.