ISSN:0975-9646

Implementing REA Algorithm to Increase Performance of the Encrypted Databases While Query Processing

Rohini A. Chirde, Prof. S. S. Kulkarni

PRMIT & R, Badnera-Amravati (M.S.)

***Abstract*—Protecting data is at the heart of many secure systems and many users depend on a database management system (DBMS) to manage the protection. There is substantial current interest in DBMS security because databases are newer than programming and operating systems. Databases are essential to many business and government organizations,holding data that reflect the organization's core competencies. Thus databases are more than software-related repositories. Their organization and contents are considered valuable corporate assets that must be carefully protected.Encryption is a science of secret writing. It is the art of protecting the information by transforming it into an unreadable format in which a message can be concealed from the casual reader and only the intended recipient will be able to convert it into original text. Encryption is becoming the last line of defense in database management system(DBMS)security. There is the essentiality of algorithms which are protected from intruder as well as maintain the performance of the system. Here we present a solution,a new encryption algorithm "Reverse Encryption Algorithm(REA) with its implementation details and examine its benefits over other algorithm.**

***Index words*—Protection,encryption,database.**

I. INTRODUCTION

What is Sensitive data ?

Many databases contain the sensitive data. We can define the sensitive data as the data that should not be made public. Some databases like public library catalog contain no sensitive data. Other databases such as defence related ones are completely sensitive. These are the two types-nothing sensitive and everything sensitive- are easiest to handle. The more difficult situation ,which is also the more interesting one is the type in which some but not all data is sensitive.We will examine this type of database.

You can take an active approach to making sure your information has not falleninto the hands of those who would misuse it forfinancial gain or other reasons.Encryption in database systems is an important issuefor research, as protected and efficient algorithms are essential that provide the ability to query over encrypted database and allow optimized encryption and decryption of data..This paper ensures maximum protection and limits theadded time cost for encryption and decryption so as to not decrease the performance of a database system.

However, how tomake queries efficiently over the encrypted database becomes a challenge. This usually implies that the system has to give up the performance to obtain the security. When data is stored in the encrypted form, we have to decrypt all the encrypted data before querying them. It is impractical solution because the cost of decryption over all the encrypted data is very expensive.

So a efficient system is needed which will provide the protection as well as provide good performance with reduced time cost for encryption and decryption.

The prescribed encryption algorithm, known as "Reverse Encryption Algorithm (REA)" is efficient and secure.

II.CONCEPT OF A DATABASE

A database–management system(DBMS) is a collection of data and a set of programs to access those data. The collection of data,usually referred to as the database,contains information relevant to an enterprise. The primary goal of a DBMS is to provide a way to store and retrieve database information that is both convenient and efficient.

Database system Applications :

- **Banking:** For customer information ,accounts,loans and banking transactions.
- **Airlines:**For reservations and schedule information.
- **Human resource:**For information about employees,salaries,paychecks.
- **Universities:**For student information,course registration and grades.
- **Credit card transactions** : For purchase on credit cards.
- **Telecommunications :**For keeping records of calls made,maintaining balances on prepaid calling cards.
- **Finance :** For storing information about holdings,sales and purchase of stocks and bonds.

III. ENCRYPTION

Encryption is a mechanism that protects your valuableinformation, such as your documents, pictures.

Encryption can be defined as: "The conversion of plaintext or data into unintelligible form by means of a reversible translation, based on a translation table or algorithm." In other words, encryption is the process of taking some data (usually called plaintext) and obfuscating it (usually with the use of some key) so that it cannot be read by others (those who do not have the correct key). The resultant data is usually called ciphertext. Conversely, the processes of taking the ciphertext and decrypting it back to its original form (usually using a key) is called decryption. Thus, decryption is defined as: "The translation of encrypted text

or data (called ciphertext) into original text or data (called plaintext)."

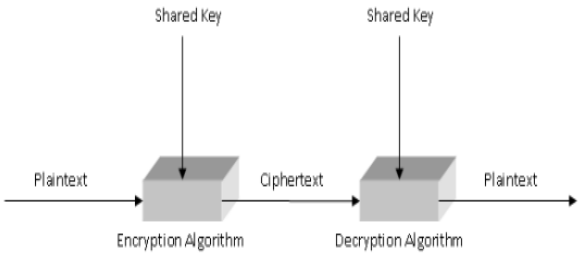


Fig 1 : For Encryption and Decryption

For many organizations, databases are a treasure trove of sensitive information containing data ranging from customers' personal details and confidential competitive information to intellectual property. Lost or stolen data, especially customer data, can result in brand damage, competitive disadvantage and serious fines.In high-profile cases, compromised data presents organizations with long-term customer acquisition and retention difficulties.

As a result, database security is a top priority for today's IT director. Yet, the shortcomings of many traditional database security techniques such as firewalls and application security have been exposed in recent years and it is now broadly recognized that these approaches to database security are no longer sufficient to protect businesses and data in today's modern, open and complex IT environment. In trying to mitigate the risk of security breaches and to comply with numerous existing and emerging regulations, database encryption is often seen as the solution. Encryption often heralded as the best defence against database security breaches.

It's clear that in certain industries which handle particularly sensitive data, such as financial services and government, regulation has emerged as the true driving force for the increased use of encryption.

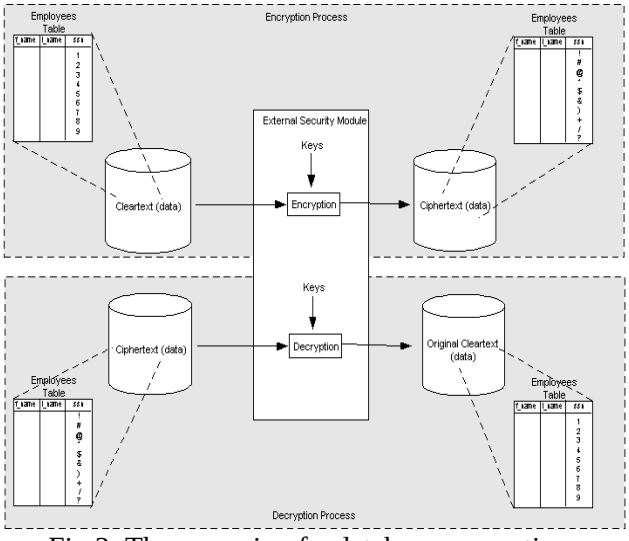


Fig 2 :The scenario of a database encryption.

IV. THE AES ENCRYPTION ALGORITHM

The AES is likely to be the commercial-grade symmetric algorithm of choice for years. Let us look at it more closely.

In January 1997,NIST called for cryptographers to develop a new encryption system. As with the call for candidates from which DES was selected,NIST made several important restrictions. The algorithms had to be

- Unclassified
- Publicly disclosed
- Available royalty-free for use worldwide
- Symmetric block cipher algorithm,for blocks of 128 bits
- Usable with key sizes of 128,192 and 256 bits

The AES was adopted for use by the U.S. government in December 2001 and became Federal Information Processing Standard 197[NIS01]. It supersedes the Data Encryption Standard (DES),which was published in 1977

Structure of the AES

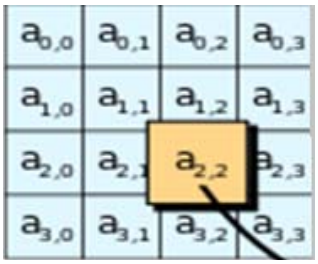
AES is a block cipher of block size 128 bits.The key length can be 128,192, or 256 bits

The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the ciphertext. The number of cycles of repetition are as follows:

•10 cycles of repetition for 128-bit keys.
•12 cycles of repetition for 192-bit keys.
•14 cycles of repetition for 256-bit keys.

It is convenient to think of a 128-bit block of AES as 4×4 matrix,called the "state"

We present the state here as the matrix s[0,0]..s[3,3]. The state is filled from the input in columns. Assume ,forexample,that the input is the 16 bytes a_{00},a_{01},a_{02}



The four steps of the algorithm operate as follows :

1.Byte substitution: The first step is a simple substitution: s[i,j] becomes s'[i,j], through a defined substitution table.

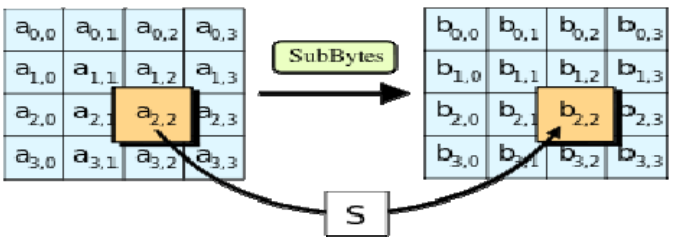


Fig : In the SubBytes step, each byte in the state is replaced with its entry in a fixed 8-bit lookup table, S;bij = S(aij).

2. Shift row : In the second step,the rows of s are permuted by left circular shift; the first(leftmost,high order) I elements of row I are shifted around to the end (rightmost,low order).

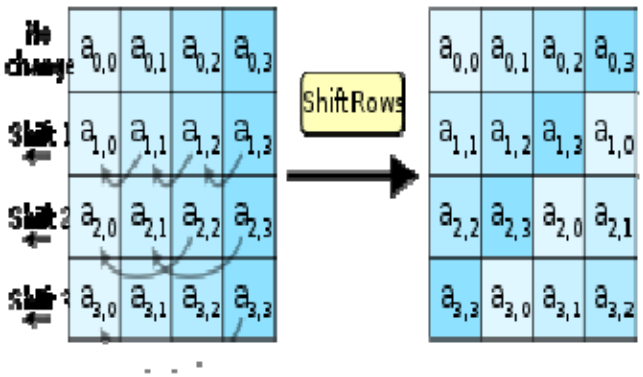


Fig : In the ShiftRows step, bytes in each row of the state are shifted cyclically to the left. The number of places each byte is shifted differs for each row

3. .Mix columns : The third step is a complex transformation on the columns of s under which the four elements of each column are multiplied by a polynomial;essentially diffusing each element of the column over all four elements of that column.

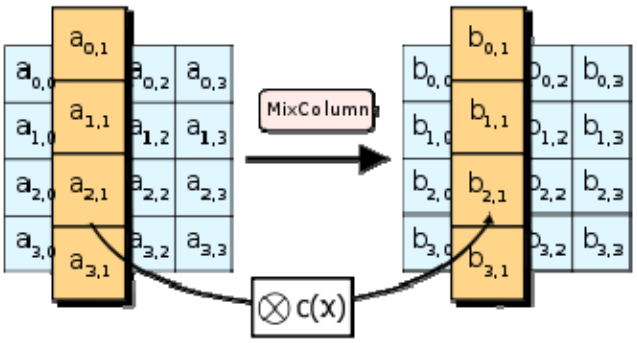


Fig : In the MixColumns step, each column of the state is multiplied with a fixed polynomial *c(x)*.

4. Add round key : Finally, a key is derived and added to each column.

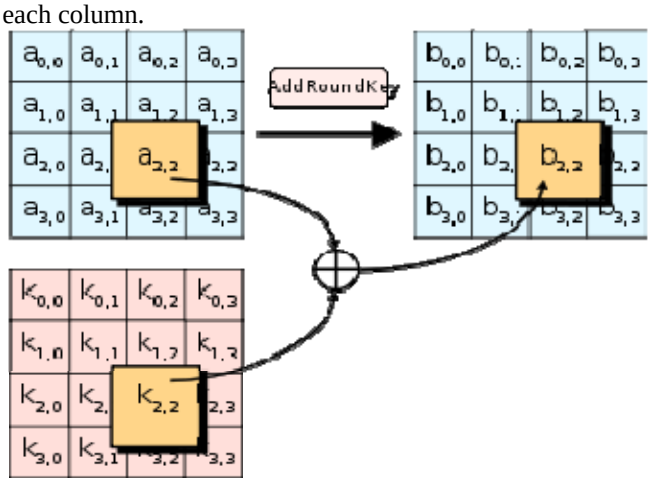


Fig : In the AddRoundKey step, each byte of the state is combined with a byte of the round subkey using the XOR operation (⊕).

V. THE PROPOSED ENCRYPTION ALGORITHM (REA)

We present the new encryption algorithm, "Reverse Encryption Algorithm (REA)", because of its simplecontext and efficient working. It is better than the competing algorithms. REA algorithm is lowering the added time cost for encryption and decryption so as to not decrease the performance of a database system. Our new algorithm (REA) is a symmetric stream cipher that can be effectively used for encryption and security of data. It takes a variable-length key, making it ideal for securing data.

The REA algorithm encryption and decryption consists of the same operations, only the two operations are different:

1) add the keys to the text in the encryption and removed the keys from the text in the decryption.

2) Execute divide operation on the text by 4 in the encryption and executed multiple operation on the text by 4 in the decryption.

We execute divide operation by 4 on the text to narrow the range domain of the ASCII code table at converting the text. The details and working of the proposed algorithm REA are given below.

Encryption Algorithm of the REA

We will be presenting the steps of the Reverse Encryption Algorithm REA .

Step1: Input the text and the key.
Step2: Add the key to the text.
Step3: Convert the previous text to ascii code.
Step4: Convert the previous ascii code to binary data.
Step5: Reverse the previous binary data.
Step6: Gather each 8 bits from the previous binary data and obtain the ascii code from it.
Step7: Divide the previous ascii code by 4.
Step8: Obtain the ascii code of the previous result divide and put it as one character.
Step9: Obtain the remainder of the previous divide and put it as a second character.
Step10: Return encrypted text

Decryption Algorithm of the REA

We will be presenting the steps of the decryption algorithm of the Reverse Encryption Algorithm REA

Step1: Input the encrypted text and the key.
Step2: Loop on the encrypted text to obtain ascii code of characters and add the next character.
Step3: Multiply ascii code of the first character by 4
Step4: Add the next digit (remainder) to the result Of multiplying operation.
Step5: Convert the previous ascii code to binary data.
Step6: Reverse the previous binary data.
Step7: Gather each 8 bits from the previous binary data and obtain the ascii code from it.
Step8: Convert the previous ascii code to text.
Step9: Remove the key from the text.
Step10: Return decrypted data

Figure for Encryption :

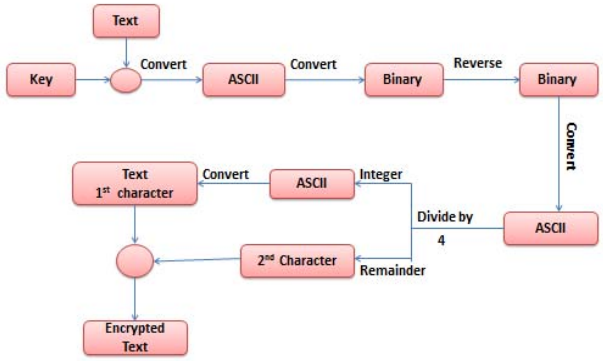


Fig.1 : Steps of the REA encryption algorithm

Figure for Decryption :

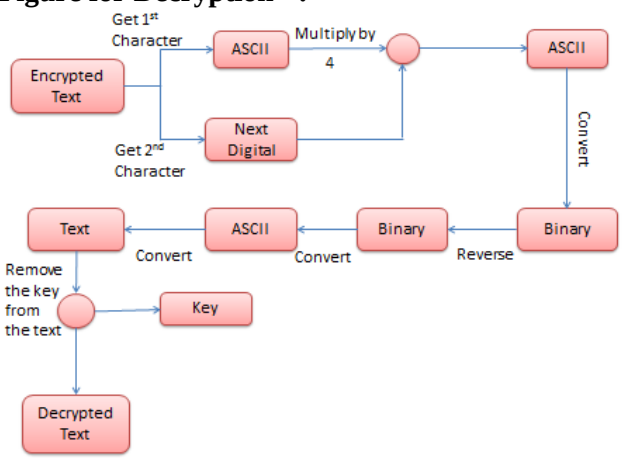


Fig.2: Steps of the REA decryption algorithm

REA: An Examples Cipher

We have example on which we have applied our new encryption algorithm REA:

1) Text to explain the running methods the proposed algorithm REA.

Text:

The first example on which we applied our new encryption algorithm REA is on the text, the explanation has been provided below.

The text is : Welcome! to PRMIT & R.
The key is: "rac" (It takes a variable-length key)
Encrypted text is:
323130*0&2$3'0$0$2&27273,2$0"2#0'232122021

Encryption:

1) Add the key to the text:
racWelcome! to PRMIT & R.

2) Convert the previous text to ascii code.
r --> 114, a --> 97, c --> 99, W --> 87, e --> 101, …….

3) Convert the previous ascii code to binary data:
00110001 00110010 00110011 01010111 01100101…..

4) Reverse the previous binary data:
11001110 11001101 11001100 10101000 10011010..…

5) Gather each 8 bits from the previous binary data and obtain the ascii code of it:
206 205 204 168 154.....

6) Divide the previous ascii code by 4 and obtain the ascii of the result(put it as one ascii character) and obtain the remainder (put it as second character).
- 206/4 = 51 ---> 3 and the remainder (next digit) = 2 (put its as 32).
- 205/4 = 51 ---> 3 and the remainder (next digit) = 1 (put its as 31).
- 204/4 = 51 ---> 3 and the remainder (next digit) = 0 (put its as 30).
- 168/4 = 42 ---> * and the remainder (next digit) = 0 (put its as *0).
- 154/4 = 38 --->& and the remainder (next digit) = 2 (put its as &2).

7) Encrypted text is (see Figure 3):
"323130*0&2$3'0$0$2&27273,2$0"2#0'232122021"

Decryption:

1) Loop on the encrypted text to get ascii code of characters and add next character.

2) Multiply ascii code of the first character by 4 and add the next digit (remainder):
- The first character = 3 --->ascii code is: 51 and the next digit(remainder)= 2 then new ascii code is: 206 = 51*4+2
- The first character = 3 --->ascii code is: 51 and the next digit(remainder)= 1 then new ascii code is: 205 = 51*4+1
- The first character = 3 --->ascii code is: 51 and the next digit(remainder)= 0 then new ascii code is: 204 = 51*4+0
- The first character = * --->ascii code is: 42 and the next digit(remainder)= 0 then new ascii code is: 168 = 42*4+0
- The first character = & --->ascii code is: 38 and the next digit(remainder)= 2 then new ascii code is: 154 = 38*4+2

3) Convert final ascii code to binary data:
11001110 11001101 11001100 10101000 10011010..…

4) Reverse the previous binary data:
00110001 00110010 00110011 01010111 1100101…..

5) Convert binary data to ascii code and text:
114 97 99 87 101 …..

6) Remove the key from text:
racWelcome! to PRMIT & R.

7) Decrypted text is (see Figure 4):
" Welcome! to PRMIT & R.

VI. Proposed System Implementation

Requirements and Input Output Specification

Software

- C#.Net, ASP. Net Microsoft SQL 2012

Operating system requirements:

- Windows XP/ Windows 7

Hardware:

- Pentium IV
- Hard Disk 120 GB
- RAM 1GB

VII. System Design

We focus on the issues that provide maximum security to database at the same time give bestperformance. Database operations are costly means that they take much time for encryption/decryption and querying of the data from the database. So we will discuss here a mechanism that will take less time as well as provide the protection to the database.

Take the three same databases.Keep the first database as it is.Encrypt the second database with AES encryption algorithm.Encrypt third database with REA encryption algorithm.Then we apply same no. of queries on the three databases.At the same time we calculate the query execution time for each algorithm. Then we will make the comparisonof the execution times of queries on the three databases and find the resultant efficient algorithm..Lastly, we will come to conclusion by obtaining the results.

n the experiments, we use three databases from the database “Rac_Bank” are:

1)Rac_Bank_Plaintext has not any encrypted fields.

2)Rac_Bank_AES has encrypted fields with AES encryption algorithm.

3)Rac_Bank_REA has the same encrypted fields in “Rac_Bank_AES” But, with using our new encryption algorithm REA.

Sample databaseforthe Rac_Bank:

Database table for account :

Account_no.	Branch_name	Balance	Date_of_deposition
432	SBI	100000	1/1/2014
678	BOI	20000	15/03/14
165	PNB	15642	20/04/14
985	BOB	56800	12/06/2014

Database table for customer :

Customer_ name	street	city	Contact_no.	Email_id
Ram	Sdk lane	Mumbai	9434555656	sdk@gmail.com
shyam	Karve road	pune	9454565768	sh@reddif.com
krishna	Tilak road	kolkata	9786564534	ks@gmail.com
bharat	Shivaji road	Amt	8734545544	bh@gmail.com

Database table for depositor :

Customer_name	Account_no.
shyam	011652365879
Ram	234598656433
krishna	098122333325
bharat	012398754442

Database table for branch :

Branch_name	Branch_street	assets
SBI	Karve road	14500000
BOI	Sdk road	23000000
PNB	Shivaji road	14780000
BOB	Tilak road	80003400

When the encryption of the fields in the databases were completed. The keys are used in the encryption were kept safe in the table encrypted withthe proposed encryption algorithm REA for the database “Rac_Bank_AES” and the database “Rac_Bank_REA”

Now, we start executing the queries on these databases. Every query from the first to fifth executes on the database “Rac_Bank_Plaintext” then calculates the execution time and repeats executes on the database “Rac_Bank_AES” then calculates the execution time and repeats the execution again on the database “Rac_Bank_REA” then calculates the execution time

A first point :the database “Rac_Bank_Plaintext” takes less time than the other databases in terms of the query execution time.Because it has no encrypted fields so it takes less time.A second point: the database “Rac_Bank_AES” is the bigger than other databases in terms of the query execution time. A third point; the database “Rac_Bank_REA” is slower than the database“Rac_Bank_Plaintext” and faster than the database “Rac_Bank_AES” in terms of the query execution time.

Chart showing comparison :

Finally, the results showed that the effects the proposed encryption algorithm REA on the database has a very good performance compared to the algorithm AES. Our new encryption algorithm REA limits the added time cost for encryption and decryption so as to not degrade the performance of the database system, if we compare with

other encryption algorithms such as AES encryption algorithm.

The proposed encryption algorithm REA represents a significant improvement over the encrypted databases. Moreover, reducing the overhead of loading on the system for the complexity of the methods that doing decryption and re-encryption the data in the database.

VIII. CONCLUSION

We present a innovative solution the Reverse Encryption Algorithm (REA) showing its benefits and mechanism over other similar encryption algorithm. We have performed the task for evaluating query processing performance over encrypted database with our new encryption algorithm (REA) and with the most widely used encryption algorithm AES.

The comparison proves the superiority of the proposed encryption algorithm REA over other encryption algorithm AES in terms of query execution time. Our new encryption algorithm REA can decrease the cost time of the encryption/decryption operations and improve the performance.

In the future , we are interested in enhancing the proposed encryption algorithm REA in order to apply it to the distributed databases.

REFERENCES

[1] Oracle, Oracle9i Database Security for eBusiness,An Oracle White Paper, June 2001.

[2] H. Hacig¨um¨us, B. Iyer, and S. Mehrotra,“Providing database as a service,” in Proceedings of ICDE, pp. 29–38, 2002.

[3] A. Ceselli, E. Damiani, S. D. C. D. Vimercati, S.Jajodia, S. Paraboschi, and P. Samarati, “Modeling and assessing inference exposure in encrypted databases,” ACM Transactions on Information System Security, vol. 8, no. 1, pp. 119–152, 2005.

[4] J. Daemen and V. Rijmen, “Rijndael: The advanced encryption standard (AES),” Dr. Dobb's Journal, vol. 26, no. 3, pp. 137-139, Mar. 2001.

[5] E. Damiani, S. D. C. D. Vimercati, M. Finetti, S. Paraboschi, P. Samarati, and S.Jajodia, “Implementation of a storage mechanism for untrusted dbmss,” IEE Security in Storage Workshop 2003, pp. 38-46, 2003.

[6] G. Davida, D. L. Wells, and J. B. Kam, “A database Encryption system with subkeys,” ACM Transactions on Database Systems, vol. 6, no. 2, pp. 312–328, 1981.

[7] E. Damiani, S. D. C. D. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, “Metadata management in outsourced encrypted databases,” in The Second VLDB Workshop on Secure Data

[8] M. R. Doomun and K. M. S. Soyjaudah, “Analytical comparison of cryptographic techniques for resource-constrained wireless security" International Journal of Network Security, vol. 9, no. 1, pp. 82-94, 2009.

[9] D. S. A. Elminaam, H. M. A. Kader, and M. M. Science, Menoufia University, Egypt in 2008. He Hadhoud, “Evaluating the performance of symmetric encryption algorithms," International Journal of Network Security, vol. 10, no. 3, pp. 213-219, 2010.

[10] S. Lee, T. Park, D. Lee, T. Nam, and S. Kim, “Chaotic order preserving encryption for efficient and secure queries on databases,” *IEICE Transactions on Information and Systems*, vol. 92, pp. 207–217, 2009.