Secured Data Storage and Retrieval using Elliptic Curve Cryptography in Cloud

Pradeep Suthanthiramani[1], Muthurajkumar Sannasy [2], Ganapathy Sannasi [3], and Kannan Arputharaj[1]
[1]Department of Information Science and Technology, Anna University, India
[2]Department of Computer Technology, Anna University, India
[3]Research Centre for Cyber-Physical Systems and School of Computer Science and Engineering, Vellore Institute of Technology, India

Abstract: *Security of data stored in the cloud databases is a challenging and complex issue to be addressed due to the presence of malicious attacks, data breaches and unsecured access points. In the past, many researchers proposed security mechanisms including access control, intrusion detection and prevention models, Encryption based storage methods and key management schemes. However, the role based access control policies that were developed to provide security for the data stored in cloud databases based on the sensitivity of the information are compromised by the attackers through the misuse of privileges gained by them from multiple roles. Therefore, it is necessary to propose more efficient mechanisms for securing the sensitive information through attribute based encryption by analyzing the association between the various attributes. For handling the security issue related to the large volume of cloud data effectively, the association rule mining algorithm has been extended with temporal constraints in this work in order to find the association among the attributes so that it is possible to form groups among the attributes as public attributes with insensitive data, group attributes with medium sensitive data and owner with highly sensitive attributes and data for enhancing the strength of attribute based encryption scheme. Based on the associations among the attributes and temporal constraints, it is possible to encrypt the sensitive data with stronger keys and algorithms. Hence, a new key generation and encryption algorithm is proposed in this paper by combining the Greatest common divisor and the Least common multiple between the primary key value and the first numeric non key attribute that is medium sensitive attributes and data present in the cloud database for providing secured storage through effective attribute based encryption. Moreover, a new intelligent algorithm called Elliptic Curve Cryptography with Base100 Table algorithm is also proposed in this paper for performing encryption and decryption operations over the most sensitive data for the data owners. From the experiments conducted in this work, it is observed that the proposed model enhances the data security by more than 5% when it is compared with other existing secured storage models available for cloud.*

Keywords: *Cloud database, secured storage, association rule mining, greatest common divisor, least common multiple, key generation and encryption.*

Received July 19, 2019; accepted June 17, 2020
https://doi.org/10.34028/iajit/18/1/7

1. Introduction

In a large transaction databases, the two widely used data analysis techniques for discovering frequently co-occurring data items and interesting association relationships between data items respectively are Association rule mining and frequent item-set mining. These two techniques have been employed in applications such as market basket analysis, health care, web usage mining, bio-informatics and prediction. In the apriori algorithm, a supermarket transaction database is taken as an example, where a transaction represents some customers and their shopping list. In this model, when a customer buys bread and butter, then he will also buy milk with high support within a time period. This type of association between the sensitive attributes can be mined through association rule mining and such associations can be used to reduce the leakage of sensitive data belonging to multiple data owners which are stored in a cloud database. Temporal constraints in association rule mining is necessary since a transaction database stored in the cloud consists of many transactions carried out by multiple data owners in a given time interval. The sensitive data of a particular data owner may easily be accessed by another data owner present in the cloud database. This leads to leakage of data which is not appropriate as per the interest of the owner and hence it is insecure with respect to storage in cloud.

Association rule mining [1] is useful to analyse the cloud data and to find the patterns present in the data. This analysis helps to find the importance of sensitive information as well as access patterns and the security level so that it is possible to perform correct partitioning of sensitive data. In a cloud database, the data are stored at the server located at the data centre. However, the important data can be partitioned either vertically or horizontally so that it is possible to replicate the important data to more than one data centre. The data owners are concerned about the security of their data because the leakage of sensitive information pertaining to one organization will enable the competing organizations to provide better services and to make policies which will affect the original organizations. This can be prevented by applying association rules for analysis and also by encrypting the data with different keys and algorithms based on sensitive levels. For example, a supermarket might gather data on customer purchasing habits. Using association rule learning, the supermarket can determine which products are frequently bought together and use this information for marketing purposes for their group of companies who supply the products to the super market. In such a scenario, it is necessary to propose effective encryption algorithm with different keys based on the rules derived from association rule mining.

The secured cloud databases are designed in such a way that they allow multiple data owners to share their data under the control of the cloud database manager securely. In a vertically partitioned cloud database, data owners wish to learn the association rules from the global database in order to separate the sensitive information from access by other users and replication to other sites. In such a scenario, the global database can be divided into a number of smaller tables in such a way that the sensitive information is separated from the ordinary attributes. In this work, new attribute based encryption algorithms are proposed with which the attributes that are used to store the highly sensitive information are encrypted using a secured key which is formed from the Greatest Common Divisor (GCD) and Least Common Multiple (LCM) values between the primary key attribute and the first numeric non key attribute of the table so that it is secured by preventing the direct access by other users. For achieving this, a new key generation, encryption and decryption algorithm called Greatest Common Divisor and Least Common Multiple (GCDLCM) key based encryption scheme has been proposed in this paper in which the encryption key is derived by combining the values of the Greatest common divisor and the Least common multiple values between the primary key value and the first numeric and sensitive column present in the sensitive database. When there is no numeric attribute is present in the database, a random number is generated by this algorithm for each tuple and this proposed algorithm is applied on these two values. This key is used to encrypt all the attribute values which are present in the sensitive table.

The major contributions of this paper are:

1. In this paper, a new secured association rule mining algorithm with temporal extensions has been proposed to identify the association between sensitive attributes in which two different keys are generated to secure the data before they are stored in the database. The keys are attached with time stamps to enhance the security further.
2. A new algorithm called GCDLCM based encryption algorithm has been proposed in this paper for generating the keys for encryption of medium sensitive data by combining the values of the Greatest common divisor and the Least common multiple values between the primary key value and the first numeric and medium sensitive column present in the database to form the key. This helps to perform the encryption more effectively using the proposed GCDLCM encryption algorithm.
3. In this model, the data are analysed using temporal and association rules obtained by the rule mining algorithm to group the data into public data, medium sensitive data (group users data) and highly sensitive (owner only data). The encryption is performed with two different algorithms where the highest sensitive data are encrypted using a special Elliptic Curve Cryptography (ECC). The medium sensitive data are encrypted with GCDLCM based encryption algorithm and the non sensitive data are stored in plain text form.
4. A new key selection and management algorithm is proposed in this paper which selects the keys and encryption method based on the sensitivity of data and the data are encrypted before they are stored in the cloud database and replicated to multiple data centres. The new key management algorithm has been designed in such a way that the highly sensitive data are encrypted using the GCDLCM based encryption algorithm proposed in this work and the most highly sensitive information are encrypted using ECC with 100 base table. Therefore, this proposed algorithm is used to encrypt the data in different security levels based on the sensitivity of attributes.
5. Finally, the database is partitioned into fragments by applying association rules with temporal constraints where the data are divided into three levels of sensitivity. They are public, group and private in which the non sensitive information are stored in plain text form which can be read by all, the information which are medium sensitive data are encrypted using the GCDLCM algorithm, followed by the new ECC algorithm called Elliptic Curve Cryptography with Base100 table (ECC-B100) algorithm that has been proposed in this paper for performing encryption and decryption operations over the highly sensitive attributes and data.

This work has been compared with the other existing algorithms for key management including chinese remainder theorem based group key management system and it is proved that the proposed algorithm increases the level of security and reduces the complexity for encryption and decryption when the key is known. The remainder of this paper is organized as follows: section 2 highlights the literature survey on the works related to the secured storage and retrieval algorithms. Section 3 presents the overall system architecture. Section 4 explains details of the proposed algorithms. Section 5 provides results and discussions. Section 6 contains the conclusion of this paper and section 7 suggests some future direction.

2. Literature Survey

There are many works in the literature, which discuss about the cloud security [14], data mining [17], data security [18, 19, 20, 21, 22, 24] and secured storage [9, 12]. Among them, Dong *et al*. [2] focused on providing security by applying integrity verification through association rules. Their verification model applies cryptographic algorithms for secured storage and retrieval. Kaosar *et al*. [6] proposed a secured and fully homomorphic encryption scheme for performing encryption process to maintain the privacy by applying association rule mining with encryption. Ge *et al*. [3] explained the issue of developing privacy preserving and distributed association rule mining in partitioned databases for enhancing the security. The major contributions of their work include collusion resistant property, effective key sharing and distributed computation.

Zhong [32] proposed new techniques for performing privacy preservation through data mining and cryptographic algorithms. Their model finds the association rules over the partitioned datasets using support and confidence measures. The major advantages of their model include the use of strong privacy using cryptographic techniques and reduction of memory using mixed fragmentation. Wong *et al*. [26] proposed security methods through the application of substitution based cipher techniques for performing effective encryption of transactional databases and to perform analysis using association rule mining. Their model reduced the vulnerabilities injected by attackers. Kantarcioglu and Clifton [5] addressed the use of security methods for partitioned databases by applying association rules over horizontally partitioned data. In their model, the authors incorporated cryptographic encryption techniques in order to minimize the leakage of shared information. Zaki [29] proposed some efficient algorithms for the effective discovery of frequent item-sets from large datasets which forms the compute intensive tasks. Misra *et al*. [13] explained about the quality of service based sensor allocation techniques for effective target tracking which is suitable in a sensor cloud environment. Their model is more applicable for sensor cloud applications. However, security is another aspect which must be considered for effective and reliable storage of information in cloud. Sarkar *et al*. [20] exploited the benefits of providing opportunistic communication techniques and efficient policies for big data management in cloud. Their model is useful for partitioning the data more effectively to perform distributed storage of the data.

Li *et al*. [9] proposed a technique for securing distributed and cloud databases. In their model, they provided a homomorphic encryption scheme and proposed a cloud based frequent item set mining algorithm to derive association rules. Their algorithm is powerful with respect to security using privacy preserving techniques. However, their model required higher runtime due to the use of rule mining and security algorithms. Therefore, it is necessary to optimize the rules derived by their proposed model for reducing the complexity. Liu *et al*. [11] developed a new authentication scheme for securing the cloud services. Their authentication scheme provided two types of authentication methods namely the local authentication scheme and the remote authentication scheme. For better authentication, new schemes with higher size key must be proposed for enhancing the security further.

Wang *et al*. [25] proposed a new privacy preserving model for transmitting the sensitive and non-sensitive data separately. Their work was demonstrated using real world datasets and they proved that their model is more secure in providing privacy preservation. Muthurajkumar *et al*. [15] proposed a new and secured Log Management Techniques with temporal features for secured storage in Cloud. The same set of authors introduced an agent based intelligent technique in the same year in another one work for detecting the malwares for infected cloud data storage files.

Modak and Shaikh [14] proposed a data hiding model on horizontally and vertically partitioned data stored in distributed databases. They used association rule mining for securing the sensitive data and distinguished the sensitive data from non-sensitive data through the application of rules. Huang *et al*. [4] proposed a secured and flexible cloud based association rule mining algorithm for providing security in horizontally partitioned distributed databases. Their work performed well than other techniques.

Yin *et al*. [28] proposed a new privacy based advanced data search scheme in which the user can generate a random query and secure the data by constructing a secured indexing method using bilinear maps. The authors conducted several experiments to prove the usefulness and strength of their model. Shen *et al*. [22] proposed a new security paradigm named remote data possession checking model with privacy preserving authenticators for securing the cloud storage. In their paradigm, both the cloud service provider and the public data verifier are not provided with access facilities since the data owner is provided with more power to handle the security of data. Dhasarathan *et al*. [1] explained about the privacy model proposed by them through the privacy preservation technique namely obfuscation for maintaining the confidentiality of information. Their model is capable of preventing digital data loss through the proposal of a powerful security mechanism. Zhang *et al*. [31] proposed a new security method called match-then-decrypt for securing the data stored in databases. In their model, the data are stored in the cipher text form which contains private details on keys along with data. The hidden access policies allow only the authorized users to read the key for decrypting the cipher text.

Zhang *et al*. [30] proposed an efficient and effective privacy preserving model for disease prediction. In their system, the patients' historical medical data have been encrypted and outsourced to the cloud server. This data is temporal in nature and hence it was used to train a classifier in order to perform accurate prediction of health conditions for patients.

In existing systems, secured association rule mining requires the display of intermediate mining results including support counts and database size in order to determine the frequent item-sets. For vertically partitioned data, the existing systems leak some of the important information due to the use of plain text form of storage. Therefore, a secured association rule mining algorithm with temporal rules is proposed in this work by extending the association rule mining algorithm with encryption and decryption techniques. The main advantage of the proposed model is that it increases the security and reduces the data leakage and decryption complexity when the keys are known. In this model, a new key management scheme which consists of two components derived from LCM and GCD was used for securing the communication between the user, the data owner and the cloud server. In this model, the cloud server allows only the data owner to store and retrieve the data by performing effective authentication checking using the proposed key management scheme.

3. System Architecture

The overall architecture of the proposed model is shown in Figure 1. It consists of nine major components namely user interfaces at n sites, local databases, knowledge base, cloud database manager, rule mining module, rule manager, security module, rule base and global database.

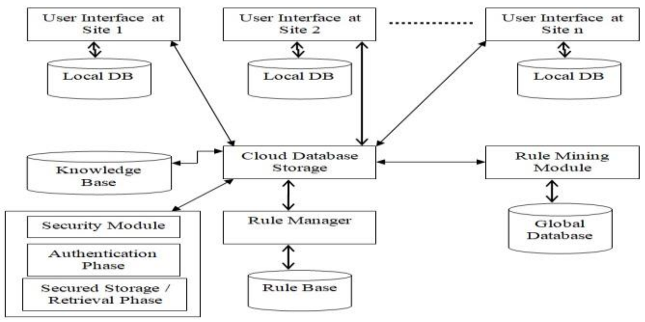

Figure 1. System architecture.

The user interface present in each of the sites is responsible for getting queries from the user, forwarding them to the cloud database manager for processing and receive the results from the database manager and to display the results to the users. The overall control of the system is with the cloud database manager and hence it is responsible for providing integrity, security, maintenance of consistency and availability of data with fast response. The knowledge base is used to store the fragmentation details, replication details, rules and storage details. The rule mining module is responsible for scanning the global database, find association rules and to inform the details to the database manager. It is also responsible to assist the database manager in fragmentation and to apply rules on them so that it is possible to divide the database into three sensitive levels and store them either in plaintext form or with encryption. The global database is responsible to store the primary copy of the data.

The security module consists of two phases such as authentication phase and secured storage/retrieval phase. Here, the authentication phase is responsible for generating keys for cloud users by applying the proposed key generation algorithm called GCDLCM algorithm. The secured storage phase is used to store and retrieve the data securely by applying the proposed algorithm called ECC-B100 algorithm. In addition, it is also responsible for making sensitive analysis with the help of the database manager and to carry out the encryption and decryption operations. The rule manager is used to store the rules, retrieve them and fire them in this knowledge base through the cloud database manager. The rule base is used to store the rules provided by the rule manager.

4. Proposed Work

In this paper, a new key generation algorithm is proposed which is the combination of Greatest common divisor and Least common multiple between the primary key value and the first numeric and sensitive column present in the sensitive database. Here, a random number is generated by this algorithm for each tuple when there is no numeric attribute which is used to find the frequent item-set and available in the sensitive table and this proposed algorithm is applied on these two values. This key is used to encrypt all the attribute values which are present in the sensitive table. Algorithm is the most important part for maintaining security since it performs effective key generation to access other data owner's data. The proposed key generation algorithm and encryption algorithm uses the Apriori algorithm [1] steps for finding frequent item set and the combination of GCD and LCM between the primary key value and the first numeric attribute value. Figure 2 demonstrates the working flow of the proposed system.

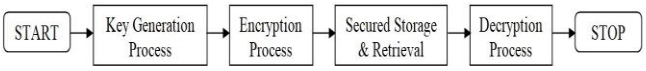

Figure 2. Flow chart.

4.1. Background

The background of the proposed model is presented in this subsection in detail. The Apriori algorithm [1] provides a sequence of steps to be followed for finding the most frequent item sets. In this work, the association rule mining algorithm has been extended with temporal constraints to find the sensitiveness of data stored at different times. The steps of the proposed Apriori algorithm with temporal extensions are presented as below:

Algorithm 1: TempARM (TDB, t1, t2)

```
Begin
  k=1;    i=0;    j=1; Max = 4;
  L[k]_Count=0;
  Read(St, Et); // Start and End times
  Read (Min_Sup, Conf);
  Call Large-Form (TDB, item _set[k]);
  While (!End_of_Data(item _set[k])) do begin
    If ( TS[k] >= St[k] && TS[k] <= Et[k]) then begin
      L[k]_Count = L[k]_Count + 1;
      i = i + 1;
      C_Set[k] = C_Set[k] U item_set[i,j];
    End;
  End;
  i= 0 ;
  For j = 2 to Max do begin
    k = k + 1;
    L[k]_Count=0;
    Call Large-Form (item _set[k]);
    While (!End_of_Data (item_set[k])) do begin
      If ( TS[k] >= St[k] && TS[k] <= Et[k]) then begin
        L[k]_Count = L[k]_Count + 1;
        i = i + 1;
        C_Set[k] = C_Set[k] U item_set[i,j];
      End;
    End;
    If (Val_seq > Min_Sup) then begin
      Extract(Val_seq, C_Set[k]);
      Prune(k, C_Set[k]);
    End;
    Form(rules[k],St[k],Et[k]);
  End;
End;
```

4.2. Key Generation

In this model, the Apriori algorithm [1, 10] with temporal analysis is used for analysing a transaction database T which uses a support threshold of e. In this paper, the association rule mining algorithm is enhanced with partitioning, key management and analysis based on sensitivity. For this purpose, a new key generation and encryption algorithm called GCDLCM algorithm is proposed in this work which is used in association rule mining algorithm for performing effective encryption of sensitive data. In this proposed algorithm, the computational complexity in the generation of keys is quantitatively reduced when compared to existing key generation techniques and it increases the security level. Moreover, the key strength remains same as that of GCD based encryption and the Chinese Reminder Theorem [25] based encryption techniques. The steps of the proposed algorithm called GCDLCM-ARM algorithm (GCDLCM-ARM) are as follows:

Algorithm 2: GCDLCM-ARM:

```
Phase 1:
N=3;
Read (ds_name, user_det);
Call auth (usr_name, key);
If authenticated = True then
  Call read (dataset);
  Call Apriori (dataset, s, c);
  For i = 1 to N do
    Form data ( item-set[i], s, c);
  Return rules (R1, R2, R3,...Rn );
  IF(sensitivity < 4 ) THEN
    T1 = T1 U TDB(Tuple)
  ELSE IF (sensitivity < 7 ) THEN
    T2 = T2 U TDB(Tuple)
  Else
    T3= T3 U TDB(Tuple)
  ENDIF
  IF (LEVEL == T2) THEN
    Call (GCDLCM,T2);
  ELSE IF (LEVEL == T3) THEN
    CALL(ECC_100,T3);
  Store (T1, T2,T3 , GDB);
  REPLICATE (T1,T2,T3);
  IF(QUERY == RETREIVE) THEN
    DECRYPT(T2, T3);
End;
```

Procedure GCDLCM ()

{

1. *Select large prime numbers p and q, where p > q from group Z_p.*
2. *Key Server: Select (k_i) from the multiplicative group Z_p for n number of users.*
3. *Now the Group Centre computes GCD and LCM values of (p,q) and finds out x_i value gcd(p,q) x lcm(p,q) = M x_i = M mod k_i*
4. *Compute y_i such that x_i x y and 1 mod k_i Compute the value x_i x y and store them in μ.*
5. *Initially, Key Server selects a random element ’k_g’ as a new Group Key within the range ’q’.*
6. *Multiply the newly generated Group Key with the value ’μ’, $g = k_g$ x μ*
7. *The Key Server broadcast a single message ’g’ to the multicast group members. On receiving ’g’ value from the Key Server, an authorized user of the current group can obtain the new Group Key k_g by doing only one mod operation, g mod $k_i = k_g$*

}

The public key is termed as “k_g” whereas the individual private key is termed as “k_i”. Each data owner with the help his individual private key deciphers the public key and finally gets the access to the database.

This algorithm is used to store the data pertaining to different data owners who are able to store and access the database in the cloud by interacting through the user interfaces provided in the n locations of the distributed cloud database. The experiments were conducted with 1000 users from 5000 locations and the results are provided in this paper. In addition, the system was first checked with a benchmark dataset and then it is also tested with real user interaction with real application data for different markets in India.

4.3. Secured Storage

This section explains in detail about the proposed Elliptic Curve Cryptography and Base100 table (ECC-B100) algorithm for performing encryption and decryption process by using the existing Elliptic Curve Cryptography (ECC) and the proposed Base100 table which is developed based on the existing base 64 table. The available data is encrypted by using Base100 table when a new key generated for the individual authorized user based on the assigned key. The encrypted form of stored data can be retrieved with the help of the key. The Base100 table is shown in Table 1.

The proposed ECC-B100 algorithm consists of two phases such as phase 1 and phase 2 for performing encryption and decryption processes. The steps of the proposed ECC-B100 algorithm are as follows:

Elliptic Curve Cryptography-Base100 Table (ECC-B100) algorithm

- *Phase* 1: Encryption

1. Read each character of the input data in sequence.
2. Change the given character from upper case to lower case and lower case to upper case.
3. Change the Space to #, $, %, & in 1st, 2nd, 3rd and 4th position of the space respectively. In case, more than 4 spaces available in the data then it will be started count from 1.
4. Compute the value of T is as the ASCII value of character and compute S=where $k_i \in Z_p$* from Key Generation part.
5. Find points (x, y) on the Elliptic Curve E $k_i(k_g, b)$. If S is odd value then P=x or If S is even value then P=y.
6. Compute Q=x_i *P // where x_i which is available in Key Generation Part and Encrypt S using Basic ECC crypto system.
7. $C_1 = y_i * P$ // where y_i is the inverse of x_i which is available in Key Generation Part.
8. $C_2 = S + (y_i * Q)$
9. Split the value of C_2 into two digit numbers.
10. C_2 is encrypted using BASE100 table.
11. Now sender sends the Encrypted message to the Receiver (C_1, C_2, x_i).

- *Phase* 2: Decryption

1. Decrypt the value of C_2 using BASE100 table.
2. Group the C_2 value which is in available in the split form with the help of comma or space as a single number (S).
3. To find the S value by using the formula S = C_2-(x_i * C_1)
4. To Compute Logk_iS = $k_g^{(\mathrm{ASCII})}$
5. To Compute ASCII = Logk_iS / k_g
6. Convert the ASCII value T into the text value.
7. Change the text value from upper case to lower case and lower case to upper case and also replace the space instead of the special characters such as #, $, % and &.
8. Decrypt the original message.

Table 1. Base100 table for encryption and decryption.

Index	Char	Index	Char	Index	Char	Index	Char
0	A	25	Z	50	«	75	n
1	B	26	`	51	»	76	o
2	C	27	;	52	□	77	p
3	D	28	:	53	ë	78	q
4	E	29	◊	54	~	79	r
5	F	30	"	55	!	80	s
6	G	31	*	56	@	81	t
7	H	32	-	57	η	82	u
8	I	33	_	58	∃	83	v
9	J	34	\	59	ϕ	84	w
10	K	35	|	60	^	85	x
11	L	36	(	61	φ	86	y
12	M	37	)	62	A	87	z
13	N	38	.	63	B	88	0
14	O	39	?	64	C	89	1
15	P	40	<	65	D	90	2
16	Q	41	>	66	E	91	3
17	R	42	[	67	F	92	4
18	S	43	]	68	G	93	5
19	T	44	{	69	H	94	6
20	U	45	}	70	I	95	7
21	V	46	≥	71	J	96	8
22	W	47	≤	72	K	97	9
23	X	48	€	73	L	98	+
24	Y	49	¶	74	M	99	/

The proposed algorithm consists of two phases such as encryption phase and decryption phase. In encryption phase, the algorithm reads each character of the input data one by one in sequence manner. Then, it takes the first character of the input data and it is considered as plain text for processing. Next, the plain text of the input data is converted from upper case to lower case and lower case to upper case. In that data, spaces are replaced for the special characters like #, $, % and & in the place of first, second, third and fourth position respectively. In case, more than four blank spaces are available in the input data then the same above mentioned special characters are to be reconsidered in the same order.

Next, it calculates the value of input data (T) is as the ASCII value of character. Then, the algorithm calculates the S value by using the key. Now, it finds the curve points (x, y) for applying in to the standard ECC formula E$k_i(k_g, b)$. Moreover, the algorithm checks the value of S is odd value. If so, then it assigns x to P y is assigned to P. After assigning the values x or *y* to P, the Q value is calculated by using x_i and P where x_i which is available in Key Generation Part. It encrypts the S value by using the standard ECC crypto system. Now, it finds the value of C_1 by applying the multiplication operation over the values P and y_i. Next, it finds the value of C_2 by adding the value S into the multiplication result of y_i and Q. Moreover, it splits the value of C_2 into two digit numbers and C_2 is encrypted using BASE100 table. Now, the sender sends the encrypted message with the values C_1, C_2, and x_i to the receiver. Table 2 shows the encryption process of the proposed algorithm.

Table 2. Example for the encryption process.

1. Extract every character 2. First, give the original character which is extracted.	G
3. Change the given character from upper case to lower case and lower case to upper case and Change the Space to #, $, %, & in 1st , 2nd, 3rd, 4th position of the space respectively.	G is converted to g. There is no Space
4. Compute T, ASCII value of character.	ASCII value of g is T = 103

5. Compute S= $k_i^{kg(ASCII)}$ where $k_i \in Zp^*$from Key Generation part.	S = 4^{206}
	S = 10,576,895,500,643,977,583,230,644,928,524,336,637,254,474,927,428,499,508,554,380,724,390,492,659,780,981,533,203,027,367,035,444,557,561,459,392,400,373,732,868,096

6. Find points (x,y) on the Elliptic Curve $E_{ki}(a,b)$.	$E_{ki}(a,b) = E_4(2,97) = E_{ki}(k_g,b)$ Points on the EC’s are, (0,1)(0,3)(1,0)(1,2)(2,1)(2,3). We choose randomly (2,3), here x=2, y=3
7. If S is odd value then P=x or If S is even value then P=y.	S is Even Number, therefore P = 3
8. To Compute Q=xi * P	$Q = x_i * P$, Q = 3 * 3 = 9
9. Encrypt S using Basic ECC crypto system	To find C1 & C2
10. C1 = y_i * P where k_g is generated in Key Generation Part.	C1 = 3 * 3, C1 = 9

11. C2 = S + (y_i* Q)	C2 = 10,576,895,500,643,977,583,230,644,928,524,336,637,254,474,927,428,499,508,554,380,724,390,492,659,780,981,533,203,027,367,035,444,557,561,459,392,400,373,732,868,096 + (3 * 9)
	C2 = 10,576,895,500,643,977,583,230,644,928,524,336,637,254,474,927,428,499,508,554,380,724,390,492,659,780,981,533,203,027,367,035,444,557,561,459,392,400,373,732,868,123

12. Split C2 into two digit numbers	C2 = 10,57,68,95,50,06,43,97,75,83,23,06,44,92,85,24,33,66,37,25,44,74,92,74,28,49,95,08,55,43,80,72,43,90,49,26,59,78,09,81,53,32,03,02,73,67,03,54,44,55,75,61,45,93,92,40,03,73,73,28,68,12,3
13. C2 is encrypted using BASE100 Table.	C2 = Kηg7«G]9nvXG{4xY_e)Z{m4m:¶7I!]sk]2¶’ϕqJtë-DCIfD~{!nφ}54<DII:gMD
Now Sender send the Encrypted message to Receiver (C1, C2, xi)	

In decryption phase, it decrypts the value of C_2 by using Basic ASCII Encode data100 (BASE100) table. Next, it groups the C_2 value which is in available in the split form with the help of comma or space as a single number (S). Then, the algorithm calculates the value k_g American Standard Code for Information Interchange (ASCII) into $\mathrm{Log}k_i\mathrm{S}$. The $\mathrm{Log}k_i\mathrm{S}$ value is divided and the resultant value is stored as an ASCII value for the text. Now, it converts the ASCII value T into the text value. The resultant text value is changed from upper case to lower case and lower case to upper case and also it replaces the space instead of the special characters such as #, $, %, and &. Finally, the decrypted message is received by the receiver in plain text form as original message. Table 3 demonstrate that the decryption process of the proposed algorithm.

Table 3. Example for the decryption process.

1. EncryptC2.	Kηg7«G]9nvXG{4xY_e)Z{m4m:¶7I!]sk]2¶’ϕqJtë-DCIfD~{!nφ}54<DII:gMD
2. Decrypt C2 using BASE100 Table.	C2 = 10,57,68,95,50,06,43,97,75,83,23,06,44,92,85,24,33,66,37,25,44,74,92,74,28,49,95,08,55,43,80,72,43,90,49,26,59,78,09,81,53,32,03,02,73,67,03,54,44,55,75,61,45,93,92,40,03,73,73,28,68,12,3
3. Group C2 is a single Numbers	C2 = 1057689550064397758323064492852433663725447492742849950855438072439049265978098153320302736703544455756145939240037373286812 3
4. S= C2 – (x_i * C1)	=1057689550064397758323064492852433663725447492742849950855438072439049265978098153320302736703544455756145939240037373286812 3 – (3 x 9) S = 1057689550064397758323064492852433663725447492742849950855438072439049265978098153320302736703544455756145939240037373286809 6
5. Compute $\mathrm{Log}_{ki}S$ = Kg(ASCII)	Log_4(1057689550064397758323064492852433663725447492742849950855438072439049265978098153320302736703544455756145939240037373286809 6) = 206
6. Get ASCII = ($\mathrm{Log}_{ki}S$)/Kg	206/2 = 103
7. Convert ASCII value T to Text Value.	T = 103 this ASCII value convert to Text value. (i.e) g
8. Change the text value from upper case to lower case and lower case to upper case and Remove the Space symbols #, $, %, & respectively.	G
We got Original Plain Text = G (Decrypted form)	

5. Result and Discussion

This system has been implemented using Amazon cloud and Java programming was used to implement the association rule mining algorithm, key management techniques and to perform secured storage and retrieval of data in order to preserve the data owner’s sensitive data in a shared database. Moreover, the proposed algorithms were tested using a benchmark dataset called the transaction dataset which is obtained from the Uniform Resource Locator (URL).

The dataset contains [27] more than 80,000 rows of Transaction Identification (TID) and the columns are the number of items [Item no] in the transaction. Since, it is a large dataset and it is split into 4 individual databases.

The frequent item-sets are generated based on the minimum support. The entire dataset is loaded and the algorithm calculates the frequent item-sets, using the transaction shown in the Tables 4 and 5. The running time for the implementation of the algorithm on each database is noted and compared.

Table 4. A's Database.

TID	Transactions
1	A1,A3
3	A2,A4
4	A3,A4
8	A1
9	A3

All the data owners run the frequent item-set mining solution to mine frequent item-sets. The frequent item-sets are calculated and encrypted supports are obtained. If data owners only want to learn the association rules, without learning the frequent item-sets, the database does not need to return these item-sets. For simplicity, the transactions for the user B are named as B1, B2,…..in order to represent the rows retrieved from the database owned by the user B.

Table 5. B's Database.

TID	Transactions
1	B1,B2,B4
3	B1,B2
4	B3
8	B2
9	B4

The joint database after pre-processing stage is shown in Table 6. It consists of five attributes namely the transaction ID, the partition owned by user A, the partition owned by user B for storing the sensitive data, the database part for A with ordinary data, and the database part for B having ordinary data.

Table 6. Joint database (after pre-processing stage).

TID	A's Partition	B's Partition	A's DB	B's DB
H(1)	S(A1) S(A3)	S(B1) S(B2) S(B4)	E(1)	E(1)
H(2)	S(A1)	S(B4)	E(0)	E(0)
H(3)	S(A2) S(A4)	S(B1) S(B2)	E(1)	E(1)
H(4)	S(A3) S(A4)	S(B3)	E(1)	E(0)
H(5)	-	S(B3)	-	E(1)
H(6)	S(A2)	-	E(0)	-
H(8)	S(A1)	S(B2)	E(1)	E(1)
H(9)	S(A3)	S(B4)	E(1)	E(1)

In this model, the proposed CGD-LCM based algorithm is used to generate a public key to be multi-casted to all the data owners in the shared database and a unique private key is unicasted to all the individual data owners separately. Later, the data owners use their respective private key to decrypt the public key and get permission to access other data owner's sensible data.

Table 7 shows the time complexity analysis between the proposed model and the existing cryptographic algorithms such as GCD and Chinese Reminder Theorem.

Table 7. Time complexity analysis.

Algorithms		
GCD	Chinese Reminder Theorem	Proposed Model
O(n)	O(n log n)	O(n^2)

From Table 7, it can be observed that the proposed model is better than the existing algorithms due to the use of GCD and LCM with respect to increase in time complexity. For a malicious user who is attempting to break the key, the proposed model needs more operations than the GCD based model and the Chinese Remainder theorem based security models.

Figure 3 shows the security level analysis for the proposed secured storage model and the existing cryptographic algorithms. Here, we have conducted five different experiments such as E1, E2, E3, E4, and E5.

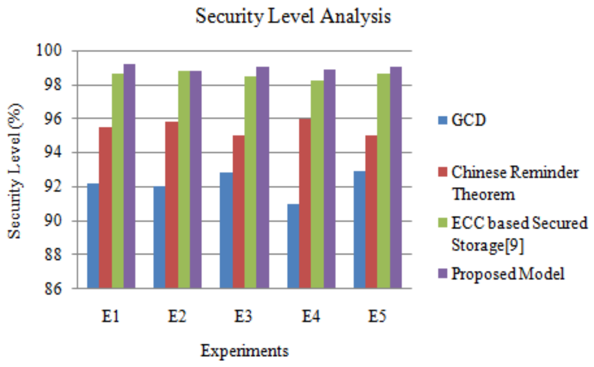

Figure 3. Security analysis for storage.

From Figure 3, it can be observed that the performance of the proposed model is better when it is compared with the existing algorithms such as GCD, Chinese Reminder Theorem and the CRT based Secured Storage explained in Cloud [7]. This performance improvement is due to the use of stronger key generation algorithm and the encryption using the proposed ECC-B100 algorithm.

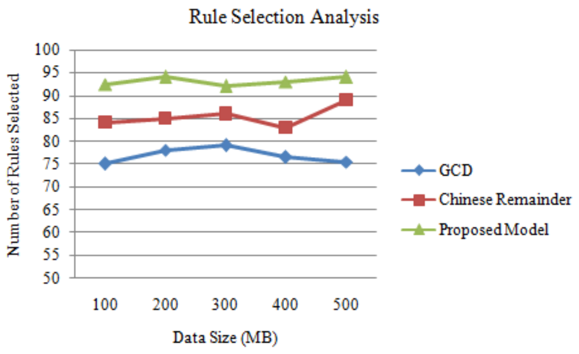

Figure 4. Rule selection analysis.

Figure 4 shows the number of rules which are encrypted with private key using the proposed model and the existing models to generate the keys based on the size of the rules to be encrypted for different data sizes.

From Figure 4, it can be observed that the proposed model selected more number of rules for analysing the sensitive data when it is compared with the other existing models namely the GCD based model and the Chinese Remainder theorem based model. The increase in the number of rules provides a better model for security analysis. Moreover, the proposed algorithm performs the necessary encryption using the GCDLCM algorithm for making a key for encryption and decryption when the sensitiveness is medium.

Figure 5 shows the time analysis between the proposed encryption model with association rule mining and the existing homomorphic encryption models proposed by Dong *et al*. [2] and Li *et al*. [9]. We have conducted five different experiments which were repeated with different size of data selected from the dataset.

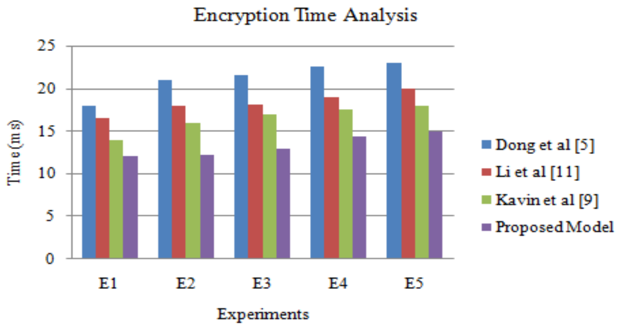

Figure 5. Encryption time analysis.

From Figure 5, it can be observed that the encryption and decryption time is less for the proposed model when it is compared with the existing techniques namely the model proposed by Dong *et al*. [2], the model proposed by Li *et al*. [9] and the ECC based secured storage model in cloud proposed by Kavin *et al*. [7]. This is due to the fact that the key generation uses temporal constraints more effectively.

Figure 6 shows the data retrieval time analysis between the proposed model and the existing models such as Two Round Searchable Encryption (TRSE), Single Searchable Encryption (SSE) and Pratiba *et al*. [17].

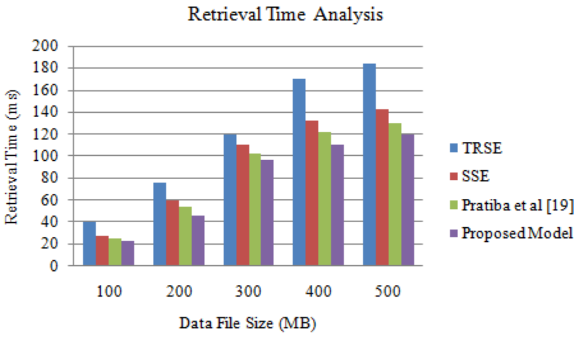

Figure 6. Data retrieval time analysis.

From Figure 6, it can be observed that the performance of the proposed model is better when it is compared with the other existing schemes in this direction in terms of time taken. Moreover, it takes less amount of time for retrieving the data from the cloud databases. This is due to the use of the effective cryptographic encryption algorithms such as GCDLCM and ECC-B100 algorithms for the attributes with different sensitive levels.

6. Conclusions

In this paper, a new temporal association rule mining algorithm using apiriori algorithm and two encryption algorithms namely GCDLCM and ECC with base 100 table has been proposed for secured storage of cloud data. The time complexity of the proposed model is O(n^2) and proved as better in terms of encryption, decryption, data retrieval time analysis. From the experiments conducted in this work, it is proved that the proposed model enhances the security when it is compared with the existing algorithms for cloud security.

7. Future Work

Future works in this direction could be the introduction of intelligent agents for effective decision making over the key generation process.

References

[1] Dhasarathan C., Thirumal V., and Ponnurangam D., "A Secure Data Privacy Preservation for on-Demand Cloud Service," *Journal of King Saud University-Engineering Sciences*, vol. 29, no. 4, pp. 144-150, 2017.

[2] Dong B., Liu R., and Wang H., "Result Integrity Verification of Outsourced Frequent Itemset Mining," *in Proceedings of IFIP Annual Conference on Data and Applications Security and Privacy*, Newark, pp. 258-265, 2013.

[3] Ge X., Yan L., Zhu J., and Shi W., "Privacy-Preserving Distributed Association Rule Mining based on the Secret Sharing Technique," *in Proceedings of the 2nd International Conference on Software Engineering and Data Mining*, Chengdu, pp. 345-350, 2010.

[4] Huang C., Lu R., and Choo K., "Secure and Flexible Cloud-Assisted Association Rule Mining over Horizontally Partitioned Databases," *Journal of Computer and System Sciences*, vol. 89, pp. 51-63, 2017.

[5] Kantarcioglu M. and Clifton C., "Privacy-Preserving Distributed Mining of Association Rules on Horizontally Partitioned Data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 9, pp. 1026-1037, 2004.

[6] Kaosar M., Paulet R., and Yi X., "Secure Two-Party Association Rule Mining," *in Proceedings of the 9th Australasian Information Security Conference*, Perth, pp. 15-22, 2011.

[7] Kavin B., Ganapathy S., Kanimozhi U., and Kannan A., "An Enhanced Security Framework for Secured Data Storage and Communications in Cloud Using ECC, Access Control and LDSA," *Wireless Personal Communications*, vol. 115, pp. 1107-1135, 2020.

[8] Krishnamurthy M., Kannan A., Baskaran R., and Kavitha M., "Cluster based Bit Vector Mining Algorithm for Finding Frequent Item sets in Temporal Databases," *Procedia Computer Science*, vol. 3, pp. 513-523, 2011.

[9] Li L., Lu R., Choo K., Datta A., and Shao J., "Privacy-Preserving-Outsourced Association Rule Mining on Vertically Partitioned Databases," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 8, pp. 1847-1861, 2016.

[10] Lian H., Wang Q., and Wang G., "Large Universe Ciphertext-Policy Attribute-Based Encryption with Attribute Level User Revocation in Cloud Storage," *The International Arab Journal of Information Technology*, vol. 17, no. 1, pp. 107-117, 2019.

[11] Liu H., Ning H., Yue Y., Wan Y., and Yang L., "Selective Disclosure and Yoking-Proof based Privacy-Preserving Authentication Scheme for Cloud Assisted Wearable Devices," *Future Generation Computer Systems*, vol. 78, no. 3, pp. 976-986, 2018.

[12] Liu X., Deng R., Yang Y., Tran H., and Zhong S., "Hybrid Privacy-Preserving Clinical Decision Support System in Fog-Cloud Computing," *Future Generation Computer Systems*, vol. 78, no. 2, pp. 825-837, 2018.

[13] Misra S., Singh A., Chatterjee S., and Mandal A., "QoS-Aware Sensor Allocation for Target Tracking in Sensor-Cloud," *Ad Hoc Networks*, vol. 33, pp. 140-153, 2015.

[14] Modak M. and Shaikh R., "Privacy Preserving Distributed Association Rule Hiding Using Concept Hierarchy," *Procedia Computer Science*, vol. 79, pp. 993-1000, 2016.

[15] Muthurajkumar S., Ganapathy S., Vijayalakshmi M., and Kannan A., "Secured Temporal Log Management Techniques for Cloud," *Procedia Computer Science*, vol. 46, pp. 589-595, 2015.

[16] Muthurajkumar S., Vijayalakshmi M., Ganapathy S., and Kannan A., "Agent Based Intelligent Approach for the Malware Detection for Infected Cloud Data Storage Files," *in Proceedings of 7th International Conference on Advanced Computing*, Chennai, pp. 1-5, 2015.

[17] Pratiba D., Shobha G., and Vijayalakshmi P., "Efficient Data Retrieval from Cloud Storage using Data Mining Technique," *International Journal on Cybernetics and Informatics*, vol. 4, no. 2, pp. 271-279, 2015.

[18] Ramachandran C., Obaidat M., Misra S., and Pena-Mora F., "A Secure and Energy-Efficient Scheme for Group-based Routing in Heterogeneous Ad Hoc Sensor Networks and its Simulation Analysis," *Simulation*, vol. 84, no. 2-3, pp.131-146, 2008.

[19] Sahoo C. and Goswami V., "Cost and Energy Optimisation of Cloud Data Centres through Dual VM Modes-Activation and Passivation," *International Journal of Communication Networks and Distributed Systems*, vol. 18, no. 3-4, pp. 371-389, 2017.

[20] Sarkar S., Chatterjee S., and Misra S., "Evacuation and Emergency Management Using a Federated Cloud," *IEEE Cloud Computing*, vol. 1, no. 4, pp. 68-76, 2014.

[21] Sethukkarasi R., Ganapathy S., Yogesh P., and Kannan A., "An Intelligent Neuro Fuzzy Temporal Knowledge Representation Model for Mining Temporal Patterns," *Journal of Intelligent and Fuzzy Systems*, vol. 26, no. 3, pp. 1167-1178, 2014.

[22] Shen W., Yang G., Yu J., Zhang H., Kong F., and Hao R., "Remote Data Possession Checking with Privacy-Preserving Authenticators for Cloud Storage," *Future Generation Computer Systems*, vol. 76, pp. 136-145, 2017.

[23] Srikant R. and Agrawal R., "Mining Generalized Association Rules," *in Proceedings of the 21st International Conference on Very Large Database*, Zurich, pp. 407-419, 1995.

[24] Stallings W., *Cryptography and Network Security*, Principles and Practices. USA: Prentice Hall, 2006.

[25] Wang W., Chen L., and Zhang Q., "Outsourcing High-Dimensional Healthcare Data to Cloud with Personalized Privacy Preservation," *Computer Networks*, vol. 88, pp. 136-148, 2015.

[26] Wong W., Hung E., Kao B., Cheung D., and Mamoulis N., "Security in Outsourcing of Association Rule Mining," *in Proceedings of the 33rd International Conference on Very Large Data Bases*, Vienna, pp. 111-112, 2007.

[27] Viger P., Lin J., Gueniche G., Deng A., and Lam H., "The Spmf Open-Source Data Mining Library Version 2," *in Proceedings of Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Riva del Garda, pp. 36-40, 2016.

[28] Yin H., Qin Z., Ou L., and Li K., "A Query Privacy-Enhanced and Secure Search Scheme over Encrypted Data in Cloud Computing," *Journal of Computer and System Sciences*, vol. 90, pp. 14-27, 2017.

[29] Zaki M., "Scalable Algorithms for Association

Mining," *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, no. 3, pp. 372-390, 2010.

[30] Zhang C., Zhu L., Xu C., and Lu R., "PPDP: An Efficient and Privacy-Preserving Disease Prediction Scheme in Cloud-Based E-Healthcare System," *Future Generation Computer Systems*, vol. 79, no.1, pp. 16-25, 2018.

[31] Zhang Y., Chen X., Li J., Wong D., Li H., and You I., "Ensuring Attribute Privacy Protection and Fast Decryption for Outsourced Data Security in Mobile Cloud Computing," *Information Sciences*, vol. 379, pp. 42-61, 2017.

[32] Zhong S., "Privacy-Preserving Algorithms for Distributed Mining of Frequent Item Sets," *Information Sciences*, vol. 177, no. 2, pp. 490-503, 2007.

Pradeep Suthanthiramani is pursuing Ph.D in DIST, CEG Campus, Anna University, Chennai, India. He has completed his M.Sc in Madurai Kamaraj University, Madurai, India. His areas of interests are cryptography and network security.

Muthurajkumar Sannasy is working as an Assistant Professor in DCT at MIT Campus, Anna University, Chennai, India. He has completed his M.E and Ph.D degrees in Anna University, Chennai. His areas of interests are including cloud security.

Ganapathy Sannasi is working as Senior Assistant Professor (Grade-II) in SCOPE and Research Center for Cyber Physical Systems at VIT-Chennai, Chennai, India. He has completed his M.E and Ph.D degrees in Anna University, Chennai. His areas of interests are including network security.

Kannan Arputharaj is working as a Senior Professor in SCOPE at VIT-Vellore, Vellore, India. He has completed his M.E and Ph.D degrees in Anna University, Chennai. He is a Retired Professor of Anna University, Chennai. His areas of interests are including data mining, cryptography and network security.