Volume 5, Issue 3, March 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Enhanced ARM with Security Using Two Keyword Playfair Cipher

[1]P. Jagannadha Varma, M.Tech, [2]T. Madhuri, [3]R. Sujatha, [4]P. Lavanya, [5]T. Avinash
[1] Assistant Professor, Dept. Of CSE, LIET, Andhra Pradesh, India
[2, 3, 4, 5] Dept. Of CSE, LIET, Andhra Pradesh, India

Abstract — With large amounts of data continuously being collected and stored in databases, many companies are interested in mining of association rules among their databases to increase their sales. FP-Growth (Frequent Pattern) algorithm is one of the ARM (Association Rule Mining) techniques for generating frequent patterns. To provide security for these obtained frequent item sets, we used modified Play Fair cipher, which is a digraph substitution technique. They provide secure transmission of message in an unreadable format and to avoid security issues which we face in existing system which uses 7x5 playfair matrix .

Keywords — Database, Association Rule Mining, FP-Growth, Substitution Technique, PlayFair Cipher.

I. INTRODUCTION

Data Mining is a process of finding hidden, non trivial and previously unknown information in large data repositories. In that Association Rule Mining (ARM) is one of the most popular data mining methods to discover interesting patterns from the data. FP-Growth algorithm is one of the ARM techniques. It is an efficient and scalable method for mining complete set of frequent patterns [11].

FP-Growth algorithm calculates all frequent item sets based on the data present in database, builds a FP-Tree structure, a compressed copy of transactional data which is stored in the storage area. All frequent item sets of articles are obtained from the FP-Tree structure. The security for these frequent item sets is provided by modified playfair cipher technique. It encrypts digraphs (pair of alphanumeric characters) instead of single character as like substitutions cipher techniques. It is comparatively very harder to break than the existing systems 7x5 play fair cipher.

II. ASSOCIATION RULES

Association analysis is used to discover interesting relations among the data that are hidden in large data repositories [2]. The undiscovered relations are represented using association rules (set of frequent items). Frequent itemsets are those items which occur in transaction frequently. For example, a customer who purchase mobile, generally makes insurance for it. These both transactions frequently appear in database of mobile store.

{Mobile} → {Insurance}

By using these type of rules retailers identifies new opportunities to sell their products.

This can be applied in application domains like web mining, bioinformatics and in industries like insurance company, online shopping and super markets. It has become essential data mining task which delivers different association rules for taking future decisions.

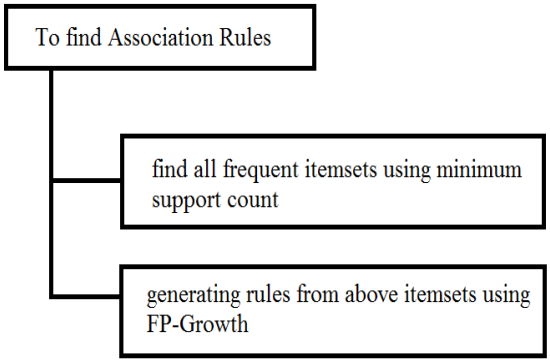


Fig 1: steps to find association rules

A. FP-Growth algorithm:

The frequent sets were generated by means of the *FP-Growth* algorithm. One of the major advantages of FP-Growth algorithm compared to other association rule mining algorithms is it uses only two scans of the data. It can also be applied on larger data sets.

FP-Growth Algorithm - FP-Tree generation

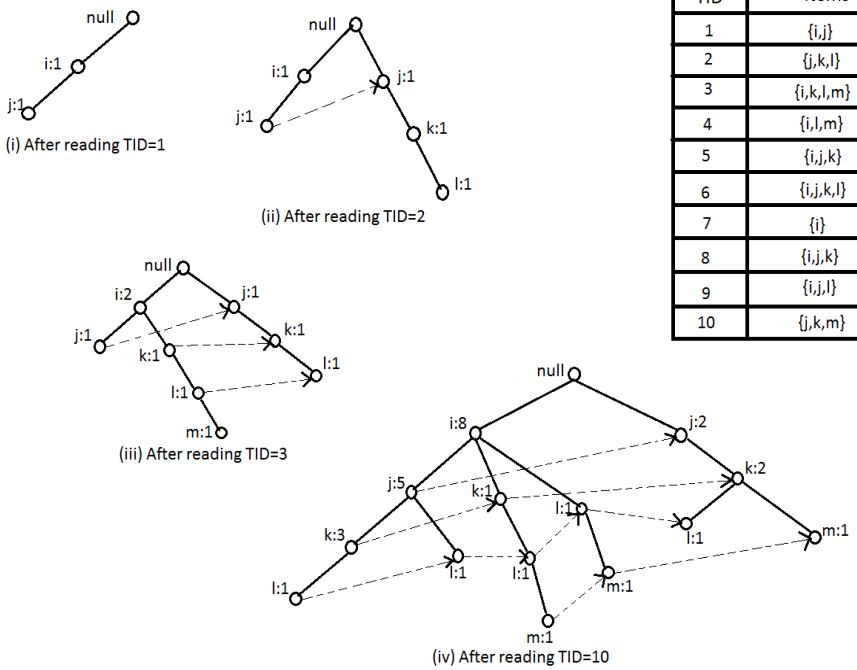
Input : D, minsupp, J ⊆I

Output : F [J] i.e.,FP-Tree


```
Algorithm     :F [J] = {};
          For all i ∈ I occurring in D {
        F [J] =F [J] ∪ {J ∪ { I }};
        //Create Di;
        Di = {};
        H = {};
    For all j ∈ I occurring in D such that j>I
        If (support(J {i, j}) ≥min supp)
        H=H ∪ {j};
        For all (tid, X) ∈ D with i ∈ X
        Di= Di ∪ {(tid , X ∩ H)};
        //Depth-first recursion
        Compute F [J ∪ { i }];
        F [J] = F[J] ∪F[J ∪ { i }];
    }
```

FP-Growth Algorithm for frequent itemset generation

```
Input       : FP-tree
Output      : total set of frequent itemsets without generating candidate item sets and     without multiple database scans.
Algorithm : FP-growth (FP-tree, null).
   Procedure FP-growth (Tree, X) {
                If Tree contains single path B
 Then for each combination (denoted as Z) of the nodes in the path B do
    Generate pattern Z ∪ X with support = min sup of nodes in Z
     Else for each xi in the header of the Tree do {
        Generate pattern Z = xi ∪ X with support = xi.support;
    Construct Z's conditional pattern base and Z's conditional FP-tree TreeZ;
        If TreeZ ≠ Ø
        Then call FP-growth (TreeZ, Z)
        }
}
```



TID	Items
1	{i,j}
2	{j,k,l}
3	{i,k,l,m}
4	{i,l,m}
5	{i,j,k}
6	{i,j,k,l}
7	{i}
8	{i,j,k}
9	{i,j,l}
10	{j,k,m}

Fig 2: Constructing FP-Tree

III. SECURING THE FREQUENT ITEM SETS

Cryptography is a science which includes study of crypting techniques for secured communication. It transforms important messages into unreadable format to get rid of security attacks. For the purpose of secure transmission of message we encrypt the message at the admin site and decrypt it at dealer site. There are two types of cryptographic techniques, symmetric-key cryptographic techniques and asymmetric-key cryptographic techniques.

In our project we use symmetric key cryptographic techniques. There are two techniques one is transposition cipher and second one is substitution cipher. Play fair cipher is a substitution cipher in which two alphanumeric characters (digraph)of the plain text is replaced by another two alphanumeric characters.

A. Existing system play fair cipher:

Existing system play fair cipher uses 7x5 data matrix grids containing keywords. The matrix is filled with the keyword (modified) and the remaining cells are filled with alphabets and numbers which are filled in alternative way. In this matrix grid alphabets (a to z) and numbers (1 to 9) are used.

7	**X**	**5**	**C**	**I**	**P**	**H**
E	**R**	A	1	B	2	D
3	F	4	G	6	J	8
K	9	L	M	N	O	Q
S	T	U	V	W	Y	Z

Keyword : 7X5CIPHER

7X5 Matrix Play Fair Cipher

Fig 3: 7x5 playfair cipher

Rules for encryption:

The 7x5 matrix must be filled up from left to right and top to bottom using the keyword (without any repetitions of characters in the keyword). There are four simple rules to create 7x5 matrix:

1) The plain text must be grouped as pair of letters and if the same letter forms a group use a filter letter (let it be x) and if there is any letter left, pair it with x.

2) If the letters fall in the same row of the matrix, replace the plain text letter with the right of it to from cipher text letter and this process continues circularly to the first letter in the row.

3) If the letters fall in the same column of the matrix, replace the plain text letter with the beneath of it to from cipher text letter and this process continues circularly to the top letter in the column.

4) If the letters are not in the same row and same column, replace the plain text letter with the letters occurring in the same row with the intersection of other plain text letter.

For example by using the above table the plain text "cipher 912 batch6" is encoded as:

Plain text : cipher912 batch6

Digraphs : CI PH ER 91 2B AT CH 6X

Cipher text: IP H7 RA MR 2D RU I7 FI

Rules for decryption:

To decrypt the cipher text ignore first rule, in rules 2 and 3 shift upwards and left side instead of downwards and right side. Rule 4 is same as encryption rule. The same keyword is used for decrypting the message.

B. 6x6 play fair cipher

Modified play fair algorithm:

Step 1: Input the two keywords

Step 2: SetKeys(keyword1,keyword2)

Step 3: KeyGeneration( )

Step 4: Encrypt (plaintext)

Step 5: Decrypt (plaintext)

The problems in 7x5 matrix play fair cipher arises when digit 0 appears in the keyword or plaintext as this technique works only with alphanumeric characters except 0 so we can't encrypt message in this situation. So in our study we proposed 6x6 play fair cipher which handles these problems efficiently. In this technique any alphanumeric string is selected as keyword.

Filling the matrix:

We first place the modified keywords in both of the matrices and the remaining cells are filled randomly with remaining alphanumeric characters as shown in fig 4 and fig 5. We encrypt plain text by using the same rules of 7x5 matrix. But the alternative pair of letters were encoded by using one keyword and the other alternative pairs of letters uses the second keyword in order to provide the fullest security for the message.

F	**A**	**I**	**R**	**Y**	**0**
1	**2**	B	V	C	P
D	5	E	6	G	7
H	L	J	9	K	Z
M	N	O	4	Q	S
T	U	3	W	X	8

Keyword : FAIRY012

6X6 Matrix Play Fair Cipher

Fig 4: play fair cipher matrix with keyword 1

P	I	R	A	T	E
4	2	0	J	L	1
D	Z	H	7	C	W
3	O	6	M	Q	U
V	K	B	5	G	9
F	Y	N	X	8	S

Keyword:PIRATE420

6X6 MATRIX PLAYFAIR CIPHER

Fig 5: play fair cipher matrix with keyword 2

The following is the example for encrypting the plain text "know your destiny".
Plain text : KNOWYOURDESTINY
Digraphs : KN OW YO UR DE ST IN Y0 3W
Cipher text: LQ UZ IQ 6E 56 8E A0 N2 WX

KN	OW	YO	UR	DE	ST	IN	Y0	3W
⇩	⇩	⇩	⇩	⇩	⇩	⇩	⇩	⇩
LQ	UZ	IQ	6E	56	8E	AO	N2	WX

Fig 6: encrypting the message.

When we decrypt the above cipher text we get the original plain text message "know your destiny" there is one to one correspondence between the plain text and cipher text so there is elimination of confusion during decryption.

IV. APPLICATION

This application can be applied to real time scenarios like in super market. The admin and dealer will have agreement. In order to the sales in the market admin wants the frequently sold items and their relationships. For this purpose he uses FP-Growth algorithm. If more than 1 rule is generated then admin will select the rule which is of his interest. For secure transmission of this rule admin encrypts it by using 6x6 play fair cipher and sends it to dealer. The dealer decrypts the message using the same play fair cipher technique. We also provide authentication for both shopkeeper and supplier (dealer).

V. SCREENSHOTS FOR THE ABOVE APPLICATION

1) Admin Login: admin logins into the system with his username and password when he wants to use it.

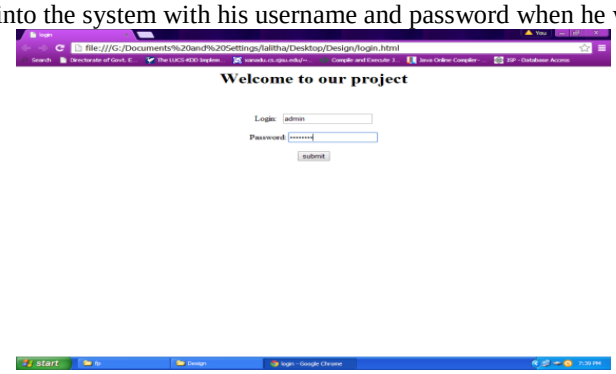


Fig 7: admin login

The tasks like inserting transactions and calculating frequent items sets, sending order were done by admin after login.

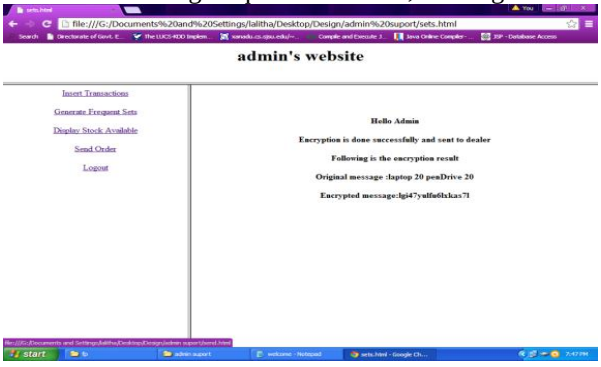


Fig 8: inserting transactions

2) When dealer wants to see the orders he will login into the system using his username and password.

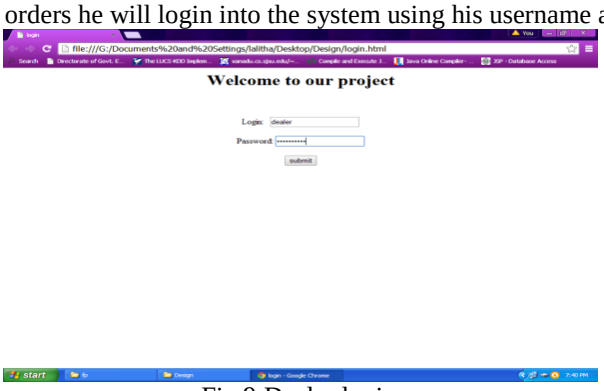


Fig 9:Dealer login

3) Dealer decrypts the orders he desired. After decrypting he dispatches the order to the admin

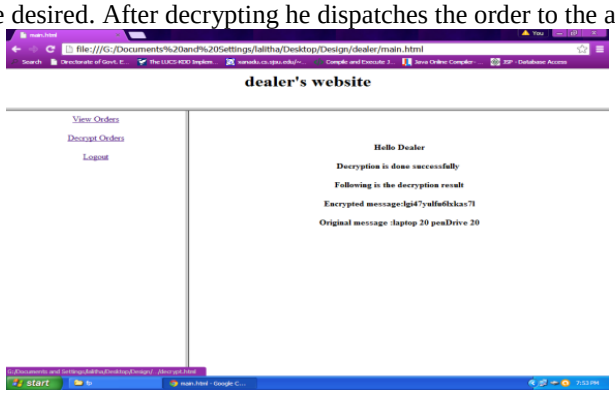


Fig 10: Dealer decrypting the order

REFERENCES

[1] P. Jagannadha Varma, Amrutha seshadri,.M. Priyanka, M.Ajay Kumar, B.L. Bharadwaj Varma.-Association Rule Mining with Security based on playfair cipher Technique - (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (1).
[2] Introduction to Data Mining Pang-Ning Tan, Michael Steinbach, Vipin Kumar
[3] en.wikipedia.org/wiki/Playfair_cipher
[4] http://technav.ieee.org/tag/496/data-mining
[5] http://www.engpaper.com/research-paper-computer-science-network-security.htm
[6] William Stallings' Cryptography and Network Security: Principles andPractice, 5e.
[7] http://en.wikipedia.org/wiki/Network_security
[8] http://en.wikibooks.org/wiki/DataMiningAlgorithms_InR/FrequentPatternMining/TheFPGrowthAlgorithm
[9] http://en.wikipedia.org/wiki/Association_rule_learning
[10] An Algorithm for Frequent Pattern Mining Based On FP-Growth oswami D.N.*,Chaturvedi Anshu Raghuvanshi C.S. SOS In Computer Science Jiwaji University Gwalior.
[11] learncryptography.com/playfair-cipher/
[12] Performance comparison of Apriori and FP-Growth algorithms in generating association rules.

