# Dual Level Security for Key Exchange using Modified RSA Public Key Encryption in Playfair Technique

Zubair Iqbal
Asst. Prof.,
Deptt. of CS & IT
Moradabad Institute of Technology
Moradabad, U.P., India

Kamal Kr. Gola
Lecturer,
Deptt. of CS & E
College of Engineering, TMU
Moradabad, U.P., India

Bhumika Gupta
Asst. Prof.,
Deptt. of CS & E
G.B. Pant Engineering College
Pauri Garhwal, U.K., India

Manisha Kandpal
Asst. Prof.,
Deptt. of ECE
Jaipur National university
Jaipur,Rajasthan, India

## ABSTRACT

The objective of this paper is to provide the dual level security for the key in playfair cipher using modified RSA digital signature scheme. This work has two phases, in the first phase the key is converted into the ASCII code and in the second phase, this work is using the concept of modified RSA digital signature scheme in which user authentication, data integrity and non-repudiation services are provided but as all know that in playfair technique, key exchange process is most important so there is a need to secure the key during key transfer that's why this work is using the concept of modified RSA public key encryption technique to encrypt the key before sending it to the receiver for decryption. This work use a 10*8 matrix which contain all alphabetic in the uppercase as well as in lowercase, the numeric value from 0 to 9, list of operator and list of bracket.

## General Terms

Key Security.

## Keywords

Playfair cipher, Plaintext, cipher text, rectangular matrix, key, encryption, decryption, public key, private key, ASCII code, Modified RSA public key encryption technique.

## 1. INTRODUCTION

Cryptography is divided into two types, Symmetric Key and asymmetric key. In Symmetric Key Cryptography a single key is shared between sender and receiver. The sender uses the shared key to encrypt the message. The receiver uses the same key to decrypt the message. While In Asymmetric Key Cryptography each user is assigned a pair of keys, public key and private key. The public key is announced to all members those are involved in the communication while the private key is kept secret by the individual user. The sender uses the public key of the receiver to encrypt the message and the receiver uses his own private key to decrypt the message[1][2].

The existing play fair cipher algorithm is based on with use of 5 X 5 matrix of letters constructed using a keyword. The existing Playfair algorithm is based on the use of a 5 X 5 matrix of letters constructed using a keyword. The 5 X 5 matrix can only allow 25 characters, hence the letters I/J count as one. If we encrypt the plain text which is having the letter I/J and when we decrypt the ciphertext at the receive end, the receiver will be under ambiguity whether to consider I or J in his text, because the meaning can be changed with the change of the letters. This algorithm can only useful for the plain text containing of alphabets but it is failed for the plain text containing of alphanumeric values. That means the plain text that is to be encrypted can only have alphabets but should not contain digits or numbers.

There are various encryption techniques in today's world. Symmetric key cryptography [3] technique is very useful for encryption process. In symmetric key cryptography, sender and receiver of a message share a single, common key that is used to encrypt and decrypt the message. Symmetric key cryptography is also called the private key cryptography. Playfair cipher [2] is one of the popular symmetric encryption methods.

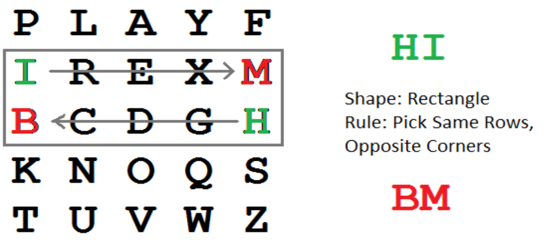

**Figure 1.1 Example of Playfair Technique**

To encrypt a message, one would break the message into digraphs (groups of 2 letters) for example; "HELLO WORLD" becomes "HE, LL, OW, OR, LD", and maps them out on the key table. If needed, append a "Z" to complete the final digraph. The two letters of the digraph are considered as the opposite corners of a rectangle in the key table [4]. Note the relative position of the corners of this rectangle. Then apply the following 4 rules, in order, to get encrypted message for each pair of letters in the PT [3].

## 2. LITERATURE REVIEW

In March 2011, Ravindra Babu K, S.Uday Kumar, A. Vinay Babu, I.V.N.S Aditya , P. Komuraiah [5] proposed a scheme based on 6 * 6 matrix in which 5 * 5 matrix has been replaced by 6 x 6 matrix. This scheme is only applicable for all the uppercase alphabets and numbers, but not applicable for the lowercase letters, white space and other special characters.

In April 2011, Shiv Shakti Srivastava, Nitin Gupta [6] proposed a scheme that shows the replacement of the 5 x 5 matrix by 8 x 8 matrix. In this work, the authors used the concept of ASCII code and seven bits binary values and for getting the cipher text, the authors used the concept of Linear Feedback Shift Register.

In May 2012, Sanjay Basu and Utpal Kumar Ray [7] proposed a scheme based on a rectangular 10 * 9 matrix which supported almost all the printable characters including the white space. In this work matrix formation is done by a secret keyword which depends on the order of placement and ciphertext is also dependent on the order of placement of different group of characters.

In January 2014, Nisarga Chand and Subhajit Bhattacharyya [8] proposed a new scheme based on 6 * 6 matrix which consists only of alphabets A to Z and numeric values 0 to 9. In this scheme the authors used for iteration steps to generate a strong encrypted message. In this scheme the authors have used four different keywords and with the help of these four keywords they encrypted and decrypted the text messages successfully.

In October 2014, Zubair Iqbal, Bhumika Gupta, Kamal Kr. Gola and Prachi Gupta [9] proposed a scheme based on 6 * 6 matrix which consists only of alphabets and numeric values. In this work, the authors used the concept of excess-3 code to generate the rectangular matrix and for ensuring the security of the key this scheme used the Caesar cipher technique.

In October 2014, Surendra Singh Chauhan, Hawa Singh and Ram Niwas Gurjar [10] proposed a scheme based on 12 * 8 matrices which consists of all alphabets, numeric values and all special characters that are used in the keyboard. In this work two phases are described, in the first phase authors increased the size of the matrix and in the second phase converted the key into ASCII code and then encrypted the key characters one by one using RSA public key encryption.

In November 2014, Kamal Kumar Gola, Bhumika Gupta and Zubair Iqbal [11] proposed a scheme to provide the data confidentiality services in the RSA digital signature algorithm. In this work the authors used the concept of public key encryption techniques to provide better security to the data (message) during data transfer.

# 3. PROPOSED ALGORITHM

## 3.1 Key Generation at the Sender Side (Public Key and Private Key)

a) Select p and q with the condition that p and q both prime and p does not equal to q.

b) Calculate n=p*q

c) Calculate Ø (n) = (p-1) * (q-1)

d) Select integer e gcd (Ø (n), e) =1; 1<e<Ø (n)

e) Calculate d e*d=1 mod Ø (n)

f) Public key (e, n)

g) Private Key (d, n)

## 3.2 Secure Key Exchange during Key Transfer

a) Select a key to encrypt the plaintext.

b) Remove the repeating characters from the key.

c) Now convert each character of the key into its equivalent ASCII code.

d) Now the sender encrypts the each ASCII code using own private key which produces the temporary encrypted key. This encrypted key is also used as a digital signature which provides the services such as user authentication, data integrity and non-repudiation.

e) After this the sender again encrypts the temporary encrypted key using the receiver's public key which produces the final encrypted key.

f) Now the sender separates the each character of the key using special character #.

g) Now the sender sends the final encrypted key to the receiver.

h) The receiver receives the key and performs the decryption process.

i) First the receiver decrypts the key using own private key which produces the temporary key.

j) The receiver again encrypts the temporary key using the sender's public key and convert that value into its equivalent ASC II code which produces the final original key.

## 3.3 Key Generation at the Receiver Side (Public Key and Private Key)

a) Select p1 and q1 with the condition that p1 and q1 both prime and p1 does not equal to q1.

b) Calculate n1=p1*q1

c) Calculates Ø (n1) = (p1) -1 * (q1) -1

d) Select integer e1 gcd (Ø (n1), e1) =1; 1<e1<Ø (n)

e) Calculate d1 e1*d1=1 mod Ø (n1)

f) Public key (e1, n1)

g) Private Key (d1, n1)

# 4. RECTANGULAR MATRIX

**Table 4.1 List of Symbol**

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **E** | **N** | **g** | **I** | **N** | **E** | **R** | **G** | **1** | **A** |
| **B** | **C** | **D** | **F** | **H** | **I** | **J** | **K** | **L** | **M** |
| **O** | **P** | **Q** | **S** | **T** | **U** | **V** | **W** | **X** | **Y** |
| **Z** | **A** | **b** | **C** | **d** | **F** | **H** | **j** | **K** | **L** |
| **M** | **O** | **p** | **Q** | **r** | **S** | **T** | **u** | **V** | **W** |
| **X** | **Y** | **z** | **0** | **2** | **3** | **4** | **5** | **6** | **7** |
| **8** | **9** | **^** | ***** | **/** | **%** | **+** | **-** | **<** | **=** |
| **>** | **!** | **\|** | **&** | **(** | **)** | **{** | **}** | **[** | **]** |

## 5. PROPOSED MODEL

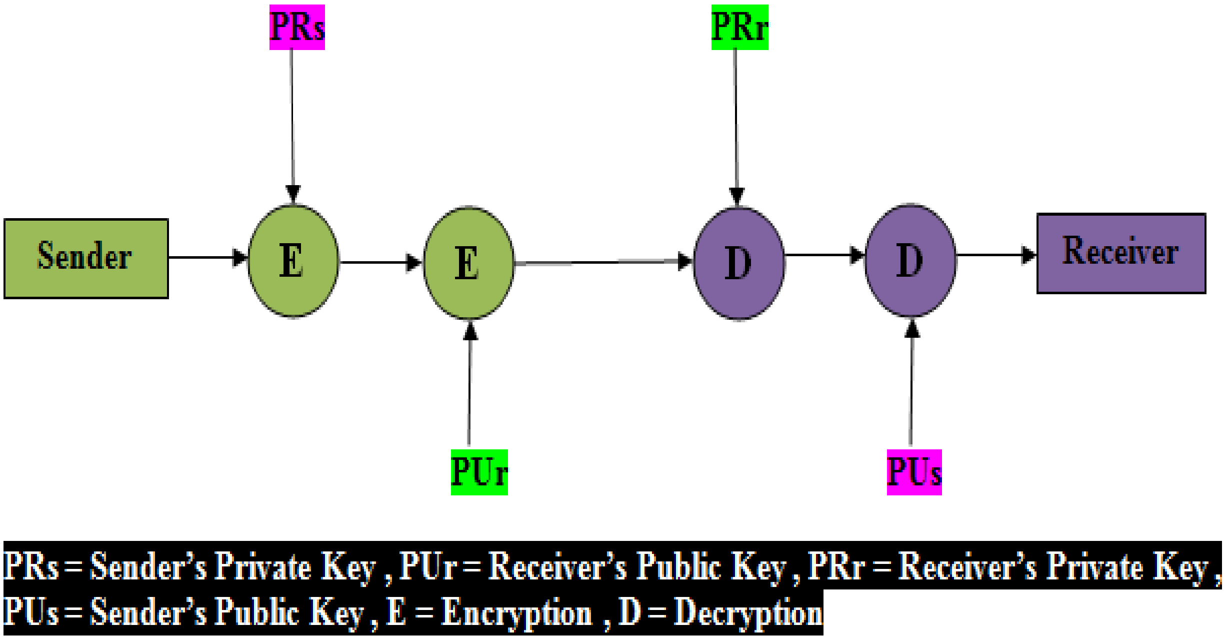
PRs
PRr
Sender
E
E
D
D
Receiver
PUr
PUs
PRs = Sender's Private Key , PUr = Receiver's Public Key , PRr = Receiver's Private Key ,
PUs = Sender's Public Key , E = Encryption , D = Decryption

**Figure 5.1 Encryption and Decryption Process**

## 6. IMPLEMENTATION

### 6.1 Key generation process (at sender side)

The first sender selects two prime numbers given as p and q that are known to sender only.

P=7 and q=17

Now it will calculate the value of n and Ø (n)

Values of n will be calculated by n=p * q.

n=7*17=119

Now calculate the value of Ø (n) = (p-1) * (q-1).

Ø (n) =6 * 16=96

Now sender will choose public key e such that e < Ø (n) and GCD (e, Ø (n)) =1.

e =5.

Now sender calculates the private key d using given expression.

e*d=1modØ (n)

(5 * d) mod 96=1

d=77.

### 6.2 Key Encryption Process

a) For implementation purpose let a key that is **EngiNeeRinG1**

b) Now remove the repeating characters from the key. Here upper case and lower case alphabets are not same. So now the key with no repeating characters is **EngiNeRG1**

c) Now convert each character into its equivalent ASCII code.

| E | N | G | I | N | E | R | G | 1 |
|---|---|---|---|---|---|---|---|---|
| 69 | 110 | 103 | 105 | 78 | 101 | 82 | 71 | 49 |

d) Now the sender encrypts the each character (one by one) of the key using own private key.

The first character is E=69

=$E^{d}$ mod n

=$69^{77}$ mod 119

= 69

The second character is n

= $110^{77}$ mod 119

= 94

The third character is g

=$103^{77}$ mod 119

=52

Fourth character is i

=$105^{77}$ mod 119

=63

Fifth character is N

=$78^{77}$ mod 119

=113

Sixth character is e

=$101^{77}$ mod 119

=33

Seventh character is R

=$82^{77}$ mod 119

=73

Eight character is G

=$71^{77}$ mod 119

=29

Ninth character is 1

=$49^{77}$ mod 119

=70

Now the table is given below

| E | N | G | i | N | E | R | G | 1 |
|---|---|---|---|---|---|---|---|---|
| **69** | **94** | **52** | **63** | **113** | **33** | **73** | **29** | **70** |

e) Now the sender again encrypts the value of the key using receiver's public key.

First value is 69

=$69^{7}$ mod 187

=86

Second value is 94

=$94^{7}$ mod 187

=19

The third value is

=$52^{7}$ mod 187

=35

The fourth value is 63

=$63^{7}$ mod 187

=24

Fifth value is 113

=$113^{7}$ mod 187

=20

Sixth value is 33

=$33^{7}$ mod 187

=33

Seventh value is 73

=$73^{7}$ mod 187

=61

Eight values is 29

=$29^{7}$ mod 187

=160

Ninth value is 70

=$70^{7}$ mod 187

=60

Now the final encrypted key is given below

| E | N | G | I | N | E | R | G | 1 |
|---|---|---|---|---|---|---|---|---|
| **86** | **19** | **35** | **24** | **20** | **33** | **61** | **160** | **60** |

f) Now insert the special symbol **#** between each character.

g) Now the encrypted key is 86#19#35#24#20#33#61#160#60

## 6.3 Key Generation Process (at Receiver Side)

Now the receiver selects two prime numbers p1 and q1 that are only known to the receiver only.

p1=17 and q1=11

Now it will calculate the value of n1 and Ø (n1)

Values of n will be calculated by n1=p1 * q1.

n1=17*11=187

Now calculate the value of Ø (n1) = (p1) -1 * (q1) -1.

Ø (n1) =16 * 10=160

Now receiver will choose public key e1 such that e1 < Ø (n1) and GCD (e1, Ø (n1)) =1.

e1 =7.

Now sender calculates the private key d1 using given expression.

e1*d1=1modØ (n)

(7 * d1) mod 60=1

d1=23.

## 6.4 Key Decryption Process

a) First the receiver receives the key and removes the special symbol **#.**

b) Now the key is

| 86 | 19 | 35 | 24 | 20 | 33 | 61 | 160 | 60 |
|---|---|---|---|---|---|---|---|---|

c) After this receiver decrypt the key using an own private key.

First value is 86

= $86^{23}$ mod 187

= 69

Second value is 19

= $19^{23}$ mod 187

= 94

Third value is 35

= $35^{23}$ mod 187

= 52

Fourth value is 24

= $24^{23}$ mod 187

= 63

Fifth value is 20

= $20^{23}$ mod 187

= 113

Sixth value is 33

= $33^{23}$ mod 187

= 33

Seventh value is 61

= $61^{23}$ mod 187

= 73

Eight value is 160

= $160^{23}$ mod 187

= 29

Ninth value is 60

= $60^{23}$ mod 187

= 70

d) Now the temporary key is

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **69** | **94** | **52** | **63** | **113** | **33** | **73** | **29** | **70** |

e) Now again the receiver decrypt the key and convert the value into its equivalent ASC II code which produce the original key.

First value is 69

= $69^{5}$ mod 119

= 69

Second value is 94

= $94^{5}$ mod 119

= 110

Third value is 52

= $52^{5}$ mod 119

= 103

Fourth value is 63

= $63^{5}$ mod 119

= 105

Fifth value is 113

= $113^{5}$ mod 119

= 78

Sixth value is 33

= $33^{5}$ mod 119

= 101

Seventh value is 73

= $73^{5}$ mod 119

= 82

Eight value is 29

= $29^{5}$ mod 119

= 71

Ninth value is 70

= $70^{5}$ mod 119

= 49

f) Now the original key is

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **69** | **110** | **103** | **105** | **78** | **101** | **82** | **71** | **49** |
| **E** | **N** | **G** | **i** | **N** | **E** | **R** | **G** | **1** |

## 7. CONCLUSION

In this paper we have analyzed the security of original Playfair cipher. This work proposed a technique to provide a dual level security to the key during transfer in playfair technique. By doing implementation we showed that proposed technique is stronger than the original Playfair cipher and provide better security to the key as compared to original playfair technique. The important concept of any algorithm satisfying security, it is one of the most important goals of the playfair technique. In the existing scheme, there is no security for the key and it does not provide the user authentication, but the proposed scheme provides key security. Also, it ensures the key confidentiality, integrity of data and user authentication. The implementation results show that the proposed method has improved the security and performance of playfair technique during key transfer, while providing high quality of service and security for desired scheme.

## 8. REFERENCES

[1] William Stallings, “Cryptography and Network Security: Principles and Practice”, 4th Edition, Prentice Hall, 2006.

[2] en.wikipedia.org/wiki/Playfair_cipher

[3] Atul Kahate, “Cryptography and Network Security”, 2nd edition, McGraw-Hill, 2010.

[4] S.S.Dhenakaran, M. Ilayaraja, “Extension of PF Cipher using 16X16 Matrix”, International Journal of Computer Applications (0975 – 888), Volume 48– No.7, June 2012.

[5] Ravindra Babu K, S. Uday Kumar, A. Vinay Babu, I.V.N.S Aditya , P. Komuraiah, “ An Extension to Traditional Playfair Cryptographic Method”, International Journal of Computer Applications (0975 ± 8887) Volume 17± No.5, March 2011.

[6] Shiv Shakti Srivastava and Nitin Gupta “A Novel Approach to Security using Extended Playfair Cipher”, International Journal of Computer Applications (0975 ±8887) Volume 20± No.6, April 2011.

[7] Sanjay Basu and Utpal Kumar Ray“Modified Playfair Cipher using Rectangular Matrix”, International Journal of Computer Applications (0975 ± 8887) Volume 46± No.9, May 2012.

[8] Nisarga Chand and Subhajit Bhattacharyya “A Novel Approach for Encryption of Text Messages Using PLAY -FAIR Cipher 6 by 6 Matrix with Four Iteration Steps” , International Journal of Engineering Science and Innovative Technology (2319-5967) Volume 3, Issue 1, January 2014.

[9] Zubair Iqbal, Bhumika Gupta, Kamal Kumar Gola and Prachi Gupta , “ Enhanced the Security of Playfair Technique using Excess 3 Code (XS3) and Ceasar Cipher”, IJCA( 0975 – 8887) Volume 103 – No 13, October 2014

[10] Surendra Singh Chauhan, Hawa Singh and Ram Niwas Gurjar, “ Secure Key Exchange using RSA in Extended Playfair Cipher Technique”, International Journal of Computer Applications (0975 – 8887) Volume 104 – No 15, October 2014

[11] Kamal Kumar Gola, Zubair Iqbal and Bhumika Gupta, “Modified RSA digital signature scheme for data confidentiality”, IJCA( 0975 – 8887) Volume 106 – No 13, November 2014