International Journal of Computing and Network Technology
ISSN 2210-1519
Int. J. Com. Net. Tech. 3, No. 3 (Sept. 2015)

Design and Analysis of Playfair Ciphers with Different Matrix Sizes

Salman A. Khan

Computer Engineering Department, College of Information Technology,
University of Bahrain, Bahrain

Received: 25 May 2015, Revised: 10 August 2015, Accepted: 20 August 2015, Published: 1 (September) 2015

Abstract: Playfair cipher is an interesting data encryption technique with a medium level of complexity, and therefore, is suitable for security of wireless and mobile systems. Different sizes of the matrices used for key have been prevsiouly stuided by researchers, but no study has been done so far that provides a comparative analysis of different matrix sizes of a key. This paper is motivated by this observation, and provides a comparison of three different matrix sizes used for the key. These matrices are 9 x 9, 10 x 10, and 11 x 11. Different experiments were performed which indicate that for the data used herein, the size of the plaintext does not have a signifncicant effect on the encryption of the output. However, the size of the key does have a notable effect and the exntrypion level increases with big keys of bigger sizes. Furthermore, increasing the size of the matrix provides better encryption results.

Keywords: Playfair cipher, Encryption, Security, Avalanche effect

1. Introduction

Cryptographic algorithms play an important role in the security architecture of any communication network. One type of these algorithms is based on symmetric key, which encrypts and decrypts data using one key. There are basically two ways to make a stronger cipher: the stream cipher, where the encryption rule is developed depending on a plaintext symbol's position in the stream of plaintext symbols, and the block cipher, which encrypts several plaintext symbols at once in a block [1].

One of the best-known early block ciphers is the Playfair system [1]. Compared to more sophisticated data encryption techniques such as RSA or DES which involve complex computational steps, Playfair ciphers are relatively less complex. From the computational and hardware point of view, more complex algorithms require more power consumption, which make them less attractive for use in wireless devices such as mobile phones, wireless sensor, etc. In contrast, Playfair cipher, being comparatively simpler than their more complex counterparts, are low power consumption algorithms and therefore are suitable for data security in wireless applications.

Playfair cipher is a symmetric encryption technique. In a symmetric encryption approach, the same key is used for encryption and decryption. Playfair cipher was invented in 1854 by Charles Wheatstone, but was named after by the name of Lord Playfair who promoted its use [2]. The technique divides the plaintext to encrypt each pair of letters (digraphs) separately, instead of single letters as in the simple substitution cipher or rather than the more complex Vigenère cipher systems.

The Playfair cipher shows great advantages over the mono alphabetic cipher. In a mono alphabetic cipher, search is done over 26 letters of the English language only, while in the Playfair cipher, the attacker has to search in 26 x 26 = 676 diagraphs. The method arranges the plaintext in a table based on a key value, where the key is usually arranged as an N x N matrix.

Some of the peculiarities of Playfair cipher are [3]:

- No plaintext letter can be represented in the cipher by itself.
- Any given letter can be represented by 5 other letters.
- Any given letter can represent 5 other letters.
- Any given letter cannot represent a letter that it combines with diagonally.
- It is twice as probable that the two letters of any pair are at the corners of a rectangle, than as in the same row or column.

A great deal of research has been done on various structures of Playfair ciphers. Cowan [4] and Mondal et al. [5] analyzed 5 x 5 ciphers. Murali and Kumar [6] also

E-mail address: sakhan@uob.edu.bh

implemented a 5 x 5 cipher using a linear feedback shift register (LFSR). Negi et al. [7] presented a design of an 8 x 8 cipher using LFSR. Hamad [8] implemented an extended 8 x 8 cipher on DNA-encoded data. Alam et al. [9] modified a 5 x 5 matrix to a 7 x 4 matrix and compared the performance of the two designs. The performance of the 7 x 4 matrix was further enhanced by Alam et al. [10]. Obayes [11] used a 5 x 5 playfair cipher in a digital steganography application. Chand and Bhattacharya [12] used a 6 x 6 matrix for encryption of text messages.

It emerges from the above discussion that, although a considerable effort has been done in analyzing Playfair ciphers of various sizes, attention has not been given to the comparative analysis and effects of different matrix sizes of the key. This observation motivates the study carried out in this paper, where key size matrices of 9 x 9, 10 x 10, and 11 x 11 are analyzed and mutually compared with respect to various features.

The rest of the paper is organized as follows. In Section II, an example is given to illustrate the functioning of a 9 x 9 Playfair cipher. Section III provides results and discussion for different experiments conducted using different input plaintext and key sizes. Finally, a conclusion is given in Section IV.

2. An illustrative example of 9 x 9 Playfair cipher

The purpose of this example is to illustrate the functioning of a 9 x 9 of playfair cipher matrix, but the concept can be extended to any size of matrix. The 9 x 9 Playfair cipher uses 9 x 9 matrix which contains the key at the beginning of the matrix. The key should not be more than 81 characters (pertaining to the size 9 x 9) decided by the security administrator. Let us take the following example to understand the functioning of the 9 x 9 Playfair cipher. In the example, the key is "playfair cipher example".

To put the key in the matrix, following steps are taken:

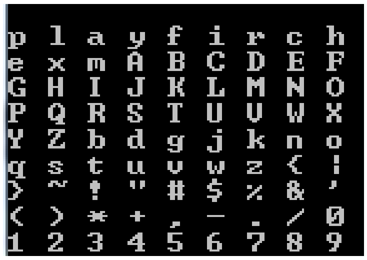
1. First, remove the spaces from the key.
2. Then, remove the duplicate character that is in the key.
3. Finally, insert the key characters at the beginning of the matrix followed by the rest of characters that are not part of the key character.

The resulting 9 x 9 matrix is depicted in Figure 1.

A. Encryption

To encrypt a given plaintext using 9 x 9 Playfair cipher, following steps are taken.

1. Replace the spaces in the plaintext by "BMW".
2. Replace the repeated letter by "AOX" and if the number of characters is odd, add "AOX" at the end of the plaintext.



```
p l a y f i r c h
e x m A B C D E F
G H I J K L M N O
P Q R S T U V W X
Y Z b d g j k n o
q s t u v w z { |
} ~ ! " # $ % & '
( ) * + , - . / 0
1 2 3 4 5 6 7 8 9
```

FIG. 1 A 9 X 9 MATRIX WITH THE KEY "PLAYFAIR CIPHER EXAMPLE"

3. Break the plaintext into digraphs (groups of 2 letters), then apply the following rules to encrypt the plaintext:
   a. If the two letters appear in the same row of the matrix, replace them with other letters that are on the right of them (if the letter is at the end of the row take the letter that is in the beginning of the row).
   b. If the two letters appear in the same column of the matrix, replace them with other letters that are below them (if the letter is in bottom of the column take the letter that is on top of the column)
   c. If the two letters are not on the same row or column, replace them with the letters on the same row respectively but at the other pair of corners of the rectangle defined by the original pair. The order is important – the first letter of the encrypted pair is the one that lies on the same row as the first letter of the plaintext pair.

As an example, consider encrypting the plaintext "My bird is on the treeLU". Replace each space by "BMW" and replace one of the duplicated letters by "AOX" as follows:

MyBMWbirdBMWisBMWonBMWtheBMWtrAOXe LU

Then, encrypt each 2 letters separately as follows:

My BM Wb ir dB MW is BM Wo nB MW th eB MW tr AO Xe LU.

Note the in the example above, BMW was used to replace the spaces in the plaintext and AOX was used to replace the repeated letter and for single last pair. The reason for doing this is the following.

- If spaces or duplicated letters are replaced by one character (e.g. X, O, ^) or any other character, but if the same character is part of the plaintext, it will be replaced by space or duplicate the next character in the decryption side.

- If we add two characters to an odd plaintext, it will result in another odd plaintext rather than even (to encrypt it as pair).

So, to solve the above problems, three different characters were used to replace the spaces in the plaintext and to replace the repeated letter and for single last pair.

Let us see how each pair in the above example plaintext will be encrypted.

1. The pair My forms a rectangle (by applying rule 3c from Section II-A), replace it with Jr (see Fig. 2).
2. The pair BM forms a rectangle (rule 3c), replace it with DK (see Fig. 2).
3. The pair Wb forms a rectangle (rule 3c), replace it with Rn (see Fig. 2).

P	L	A	Y	F	i	R	C	h
E	X	m	A	B	C	D	E	F
G	H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W	X
Y	Z	B	D	G	j	K	N	O
Q	S	T	U	V	w	Z	{	|
}	~	!	"	#	$	%	&	‘
<	>	*	+	,	_	-	/	0
1	2	3	4	5	6	7	8	9

FIG. 2 ENCRYPTING THE PLAINTEXT FOR STEPS 1 TO 4 ABOVE.

4. The pair ir is in same row (rule3 a), replace it with rc.
5. The pair MW forms a rectangle (rule 3c), replace it with NV.
6. The pair is forms a rectangle (rule 3c), replace it with lw.
7. The pair BM forms a rectangle (rule 3c), replace it with DK.
8. The pair Wo forms a rectangle (rule 3c), replace it with Xn.
9. The pair nB forms a rectangle (rule 3c), replace it with gE.
10. The pair MW forms a rectangle (rule 3c), replace it with NV.
11. The pair th forms a rectangle (rule 3c), replace it with |a.
12. The pair eB is in same row (rule 3a), replace it with xC.
13. The pair MW forms a rectangle (rule 3c), replace it with NV.
14. The pair tr forms a rectangle (rule 3c), replace it with za.

p	l	a	y	f	i	r	c	h
e	x	m	A	B	C	D	E	F
G	H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W	X
Y	Z	b	d	g	j	k	n	o
q	s	t	u	v	w	z	{	:
}	~	!	"	#	$	%	&	’
<	>	*	+	,	_	-	/	0
1	2	3	4	5	6	7	8	9

FIG. 3 ENCRYPTING THE PLAINTEXT FOR STEPS 15 TO 17 BELOW

15. The pair AO forms a rectangle (rule 3c), replace it with FJ (see Fig. 3)
16. The pair Xe forms a rectangle (rule 3c), replace it with PF (see Fig. ;23).
17. The pair LU is in same row (rule 3a), replace it with Uj (See Fig. 3).

The resulting ciphertext will be as follows:

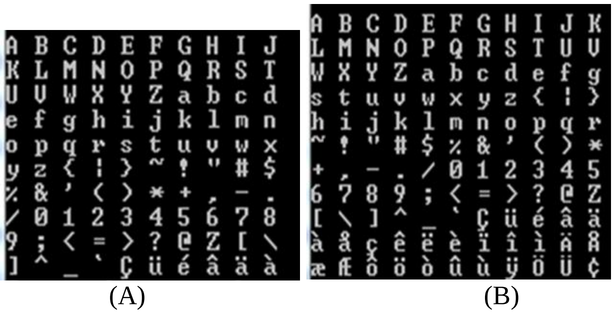
JrDKRnrcgANVlwDKXngENV|axCNVzaFJPFUj

B. *Decryption*

To decrypt the ciphertext on the receiver side, use the inverse (opposite) steps that were done in the encryption side, and the original correct plaintext will be recovered.

3. RESULTS AND ANALYSIS OF VARIOUS MATRIX SIZES FOR THE KEY

Different sizes of the matrices for the key were implemented ranging from 9 x 9 to 11 x 11. The characters contained in these matrices (without any specific key) are shown in Figures 4(A) and 4(B) for 10 x 10 and 11 x 11 respectively (note that an example of 9 x 9 is already given in Fig. 1, but with a specific key, although all 81 characters are there). Accordingly, a 10 x 10 matrix contained 100 characters, and an 11 x 11 matrix consisted of 121 characters. Two set of experiments were done. In the first set, the plaintext was variable but the key was kept the same. In the second set, the plaintext was kept the same but the key was changed. For both sets, avalanche effect was measured. Avalanche effect measures the change in the output when the input or the key is slightly changed [13]. Details of these experiments and results are given below.



(A)

A	B	C	D	E	F	G	H	I	J
K	L	M	N	O	P	Q	R	S	T
U	V	W	X	Y	Z	a	b	c	d
e	f	g	h	i	j	k	l	m	n
o	p	q	r	s	t	u	v	w	x
y	z	{	¦	}	~	!	"	#	$
%	&	'	(	)	*	+	,	-	.
/	0	1	2	3	4	5	6	7	8
9	;	<	=	>	?	@	Z	[	\
]	^	_	`	Ç	ü	é	â	ä	à

(B)

A	B	C	D	E	F	G	H	I	J	K
L	M	N	O	P	Q	R	S	T	U	V
W	X	Y	Z	a	b	c	d	e	f	g
s	t	u	v	w	x	y	z	{	¦	}
h	i	j	k	l	m	n	o	p	q	r
~	!	"	#	$	%	&	'	(	)	*
+	,	-	.	/	0	1	2	3	4	5
6	7	8	9	;	<	=	>	?	@	Z
[	\	]	^	_	`	Ç	ü	é	â	ä
à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	¢

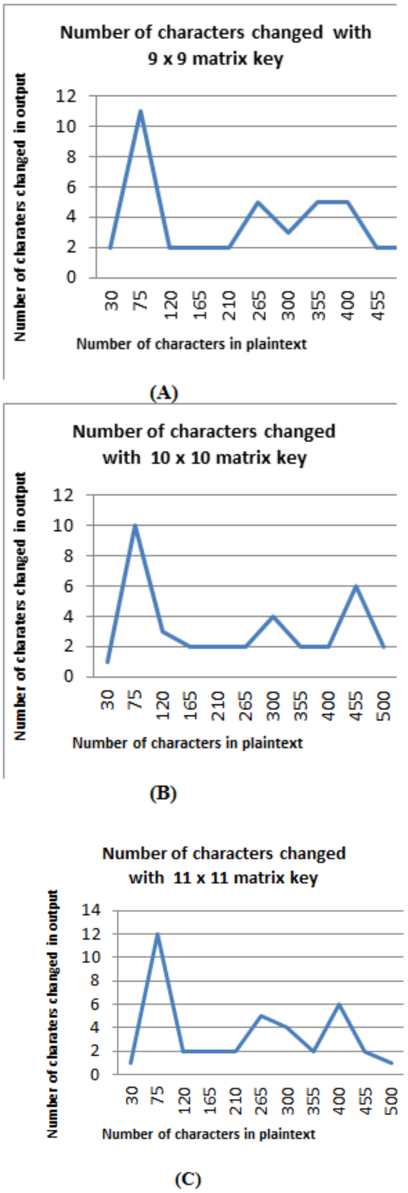
FIG 4. PLAYFAIR KEY MATRIX OF SIZE (A) 10 X 10 (B) 11 X 11

TABLE I. RESULTS FOR AVALANCHE EFFECT WITH FIXED KEY AND VARIABLE PLAINTEXT

NUMBER OF CHARACTERS IN PLAINTEXT	NUMBER OF CHARACTERS CHANGED IN OUTPUT AFTER FLIPPING ONE BIT IN INPUT		
	9 x 9	10 x 10	11 x 11
30	2	1	1
75	11	10	12
120	2	3	2
165	2	2	2
210	2	2	2
265	5	2	5
300	3	4	4
355	5	2	2
400	5	2	6
455	2	6	2
500	2	2	1

A. Results with fixed key and variable plaintexts

This set of experiments was performed with fixed matrix, fixed key (27 distinct characters including space) which is "A quick brown fox jumps over the lazy dog", and ten different sizes of plaintexts ranging from 30 characters to 500 characters. In each of these plaintexts, one bit in the input was changed (note that each character in the plaintext is represented in ASCII format, and changing one bit input anywhere will affect that particular character in which the change has taken place). Table I shows the results for different plaintexts and the corresponding changes in output characters for different matrix sizes for the key. It is observed from this table that the change in the output characters ranged between 1 and 12 characters. A clearer picture of the trends is visible in Figure 5 which shows that an increase in the size of plaintext does not contribute much to the avalanche effect. The reason for this is that there is only one bit change in the whole plaintext input, and this one bit change does not depend on the size of the plaintext. Moreover, it is also observed that for all three matrix sizes, the biggest change was observed for plaintext of 75 characters. This could be attributed to the structure of the plaintext itself, which, in this particular case, might have a strong impact on the output due to the particular key selected for encryption.



(A)

(B)

(C)

FIG 5. CHANGE IN OUTPUT WITH 1 BIT CHANGE IN PLAINTEXT INPUT USING (A) 9 X 9 MATRIX (B) 10 X 10 MATRIX (C) 11 X 11 MATRIX

TABLE II. RESULTS FOR AVALANCHE EFFECT WITH FIXED KEYAND VARIABLE

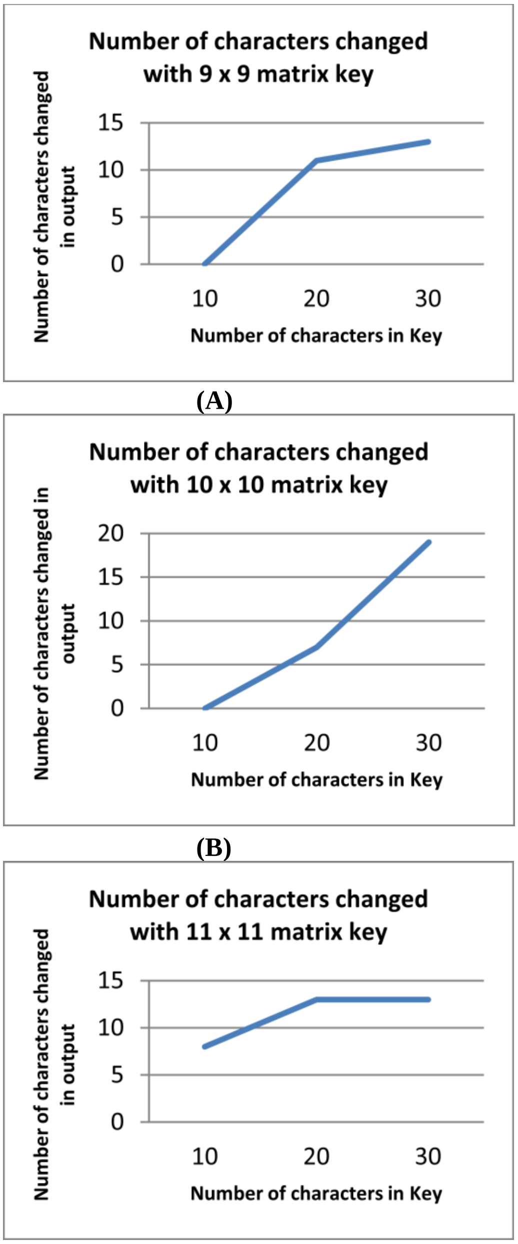
NUMBER OF CHARACTERS IN KEY	Number of characters changed in output after flipping one bit in key		
	9 x 9	10 x 10	11 x 11
10	0	0	8
20	11	7	13
30	13	19	13

B. *Results with variable key and fixed plaintexts*

The second set of experiments was having fixed plaintext (32 distinct characters including space) which was “A quick brown fox jumps over the lazy dog l,5mp?/.” Three different sizes of keys (i.e. 10, 20, and 30 characters) were used with each of configuration of the matrices. Table II shows the results of these experiments. It is observed from the table that with key size of 10 characters, there was no effect on the output for 9 x 9 and 10 x 10 matrices, but had a notable effect with 8 characters changed with 11 x 11 matrix. Moreover, when the key size was increased for any matrix size, the general trend was that the number of characters changed in the output increased sharply and steadily, with the exception of 11 x 11 matrix where change in key size from 20 to 30 and flipping one bit in the key resulted in the same effect in the output with change of 14 characters each time. The trends in Figure 6 provide a more visible illustration of the above observations. Overall, it can be fairly claimed that increasing the key size proportionally increased the number of characters changed in the encrypted output.

4. CONCLUSIONS

Playfair cipher has a strong potential for usage in wireless and mobile communications in which the sender is constrained by limited power. This potential of Playfair cipher lies in its simple design which allows for less power consumption than more complex algorithms such as RSA, DES, and AES. This paper presented a comparative analysis of three different matrix sizes, namely 9 x 9, 10 x 10, and 11 x 11, for keys of Playfair cipher. The results indicate that the avalanche effect does not depend on the size of the plaintext itself. However, there may be notable changes in the output with plaintext of certain sizes, but no change with other plaintext of the same size. That strongly depends on the plaintext itself and the order (position) of the characters it contains in the matrix. Furthermore, bigger key sizes have a stronger impact on the output compared to smaller key sizes.



(A)

(B)

(C)

FIG 6. CHANGE IN OUTPUT WITH 1 BIT CHANGE IN KEY USING (A) 9 X 9 MATRIX (B) 10 X 10 MATRIX (C) 11 X 11 MATRIX

REFERENCES

[1] R. J. Anderson, “Security Engineering: A Guide to Building Dependable Distributed Systems”, John Wiley & Sons, 2001

[2] http://en.wikipedia.org/wiki/Playfair_cipher. Retrieved on 26 May 2015

[3] S. Srivastavav and N. Gupta," Optimization and Analysis of the Extended Playfair Cipher" in Emerging Trends in Networks and Computer Communications (ETNCC 2011), Mumbai, Pages 267 - 270, April 2011.

[4] M. J. Cowan, "Breaking Short Playfair Ciphers with the Simulated Annealing Algorithm", Cryptologia, Vol. 32, pp.71–83, Taylor & Francis, 2008.

[5] U. Mondal, S. Mandal, and J. Palchodhury, "A Framework for the Developmnet of a New Apporach of Playfair Cipher", in Porceedings of IndiaCom 2008, pages 1-2, February 2008.

[6] P. Murali and G. Senthilkumar, "Modified Version of Playfair Cipher using Linear Feedback Shift Register", International Journal of Computer Science and Network Security, Vol. 8 No. 12, pp. 26-29, 2008.

[7] A. Negi, J. Farswan, V. Thakkar, and S. Ghansala, "Cryptography Playfair Cipher using Linear Feedback Shift Register", IOSR Journal of Engineering, Vol. 2, No. 5, pp. 1212-1216, 2012

[8] S. Hamad, "A Novel Implementation of an Extended 8x8 Playfair Cipher Using Interweaving on DNA-encoded Data", International Journal of Electrical and Computer Engineering, Vol. 4, No. 1, pp. 93-100, 2014.

[9] A. Alam, S. Ullah, I. Wahid, and S. Khalid, "Universal Playfair Cipher Using M x N Matrix", International Journal of Advanced Computer Science, Vol. 1, No. 3, pp. 113-117, 2011.

[10] A. Alam, B. Khalid, and C. Salam, "A Modified Version of Playfair Cipher Using 7×4 Matrix", International Journal of Computer Theory and Engineering, Vol. 5, No. 4, pp. 626-628, 2013.

[11] H. Obayes, "Suggested Approach to Embedded Playfair Cipher Message in Digital Image", Int. Journal of Engineering Research and Applications, Vol. 3, Issue 5, pp.710-714, 2013.

[12] Nisarga Chand, Subhajit Bhattacharyya, "A Novel Approach for Encryption of Text Messages Using Playfair Cipher 6 by 6 Matrix with Four Iteration Steps", International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 3, Issue 1, pp. 478-484, 2014.

Salman A. Khan received M.S. in Computer Engineering from King Fahd University of Petroleum & Minerals, Saudi Arabia in 2000 and the PhD degree in Computer Science from University of Pretoria, South Africa in 2009. He is currently an Assistant Professor in the Computer Engineering Dept. at University of Bahrain, and an Adjunct Senior Researcher with Computational Intelligence Research Group, Computer Science Department, University of Pretoria. He has published over 30 research articles in reputed journals and conferences. His research interests include Evolutionary Computation, Swarm Intelligence, Nature-inspired Algorithms, Fuzzy Logic, Single-objective and Multi-objective optimization and decision-making, Computer Networks, and Mobile Communication Systems. He serves as a reviewer for various reputed journals and conferences annually.