Message hiding in audio file using variation of playfair cipher and audio steganography

Vimalanathan P [(1)], *R Srividhya* [(2)].
[1]*M.Phil Scholars, SITS, Dr. G R D College of Science, Coimbatore.*
[2]*Assistant Professor, SITS, Dr. G R D College of Science, Coimbatore.*

Abstract

Steganography is the method by which data or files can be hidden in Image files or any other type of file. Steganography is often confused with cryptography because the two are similar in the way that they both are used to protect confidential information. The difference between the two is in the appearance of the processed output. The output of steganography operation is not apparently visible but in cryptography the output is scrambled so that it can draw attention. This paper focuses on the strength of combining cryptography and steganography methods, variation of playfair cipher, 512 base conversion which results in enhancement of security.

***Keywords** - Image Steganography, Least Significant bit (LSB), Cryptography, Audio File Tag, secret Key, Playfair cipher*

1. Introduction

The objective of steganography is to hide a secret message within a cover-media in such a way that others cannot discern the presence of the hidden message. Technically in simple words “steganography means hiding one piece of data within another”. The word "Steganography" is of Greek origin and means "covered, or hidden writing”. Modern steganography uses the opportunity of hiding information into digital multimedia files in the past people used hidden tattoos or invisible ink to convey steganographic content. Today, computer and network technologies provide easy-to-use communication channels for steganography. Steganography and Cryptography are great partners, in spite of functional difference. It is common practice to use cryptography with steganography. Naturally these techniques date back throughout history, the main applications being in couriering information during times of war. With the invention of digital audio and images files this has taken on a whole new meaning; creating new methods for performing “reversible data hiding” as it is often dubbed. This has many possible applications including the copyright watermarking of audio, video and still image data. In digital media, Steganography is mainly oriented around the undetectable transmission of one form of information within another.

The embedded data must be undetectable within its carrier medium (the audio or image file used). The carrier should display no properties that flag it as suspicious whether it is to the human visual/auditory system or in increased file size for the carrier file. The embedded data must maintain its integrity within the carrier and should be easily removable, under the right circumstances, by the receiving party. The existing system of Audio Steganography poses more restrictions on the choosing of audio files. User can select only wav files to encode. Further embedding information into sound files is generally considered more difficult than images; However human ear is extremely sensitive to perturbations in sound and can in fact detect such turbulence as low as one part in 10 million.

2. Modern techniques of steganography

The common modern technique of steganography exploits the property of the media itself to convey a message.
·Plaintext
·Image
·Audio
There are many techniques for hiding information or messages in audio in such a manner that the alterations made to the audio file are perceptually indiscemible. Common approaches Include.

i. Least significant bit (LSB)
ii. Spread spectrum (SS)
iii. Parity coding
iv. Echo technique
v. Noise Gate Technique

3. Proposed method

This paper proposes a novel method for secured data communication between 2 parties. When algorithm is adopted, user can send the file to other user then receiver is able to retrieve the message from the send file. Thus, the data is protected without revealing the content to others. The system uses 4 layers of security in order to maintain data privacy.

Algorithm

Input: Message, Key, Audio File, Image 1 & Image 2

Output: Encrypted Message in Audio

Method:

1. Key is hidden inside an audio file using LSB method.

2. Message is encrypted based upon key which uses variation of playfair cipher.

3. The encrypted message is converted to 512 based character set.

4. The Encrypted message is hidden inside an image 1 and Image 2 using LSB. Image 1 and image 2 are added to property of audio file.

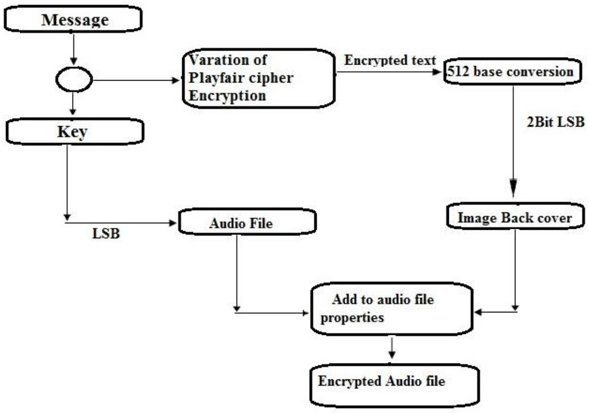


Fig. 1

3.1. LSB coding

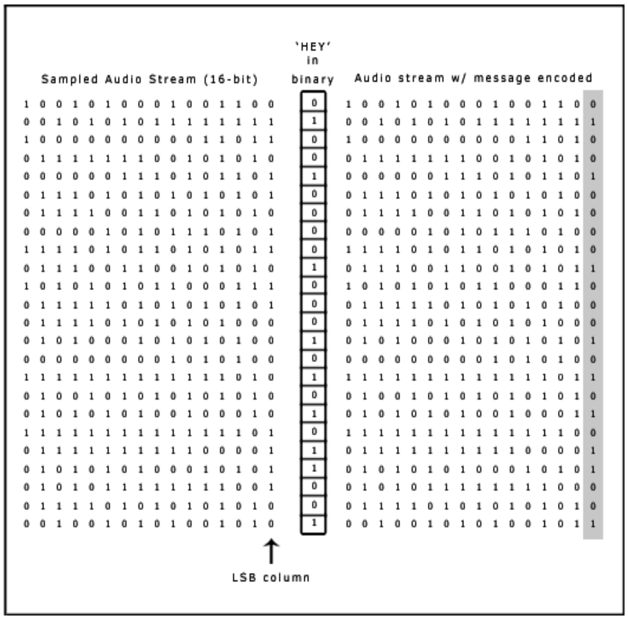
A very popular methodology is the LSB (Least Significant Bit) algorithm, which replaces the least significant bit in some bytes of the cover file to hide a sequence of bytes containing the hidden data .In this paper, LSB method is used for hiding data. It is usually an effective technique in cases where the LSB substitution doesn't cause significant quality degradation, such as in 24-bit bitmaps. In computing, the least significant bit (LSB) is the bit position in a binary integer giving the units value, that is, determining whether the number is even or odd. The LSB is sometimes referred to as the right-most bit, due to the convention in positional notation of writing less significant digit further to the right. It is analogous to the least significant digit of a decimal integer, which is the digit in the ones (right-most) position.

1	0	0	1	0	1	0	1

Figure 2: Binary representation of decimal 149.

The binary representation of decimal 149, with the LSB highlighted. The MSB in an 8-bit binary number represents a value of 128 decimal. The LSB represents a value of 1. For example, to hide the letter "a" (ASCII code 97, which is 01100001) inside eight bytes of a cover, you can set the LSB of each byte like this:

1001001**0**
0101001**1**
1001101**1**
1101001**0**
1000101**0**
0000001**0**
0111001**0**
0010101**1**



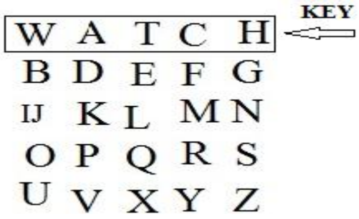
```
                                          'HEY'
                                            in
Sampled Audio Stream (16-bit)            binary   Audio stream w/ message encoded
1 0 0 1 0 1 0 0 0 1 0 0 1 1 0 0            0      1 0 0 1 0 1 0 0 0 1 0 0 1 1 0 0
0 0 1 0 1 0 1 0 1 1 1 1 1 1 1 1            1      0 0 1 0 1 0 1 0 1 1 1 1 1 1 1 1
1 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1            0      1 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0
0 1 1 1 1 1 1 1 0 0 1 0 1 0 1 0            0      0 1 1 1 1 1 1 1 0 0 1 0 1 0 1 0
0 0 0 0 0 0 1 1 1 0 1 0 1 1 0 1            1      0 0 0 0 0 0 1 1 1 0 1 0 1 1 0 1
0 1 1 1 0 1 0 1 0 1 0 1 0 1 0 1            0      0 1 1 1 0 1 0 1 0 1 0 1 0 1 0 0
0 1 1 1 1 0 0 1 1 0 1 0 1 0 1 0            0      0 1 1 1 1 0 0 1 1 0 1 0 1 0 1 0
0 0 0 0 0 1 0 1 0 1 1 1 0 1 0 1            0      0 0 0 0 0 1 0 1 0 1 1 1 0 1 0 0
1 1 1 1 0 1 0 1 1 0 1 0 1 0 1 1            0      1 1 1 1 0 1 0 1 1 0 1 0 1 0 1 0
0 1 1 1 0 0 1 1 0 0 1 0 1 0 1 0            1      0 1 1 1 0 0 1 1 0 0 1 0 1 0 1 1
1 0 1 0 1 0 1 0 1 1 0 0 0 1 1 1            0      1 0 1 0 1 0 1 0 1 1 0 0 0 1 1 0
0 1 1 1 1 1 0 1 0 1 0 1 0 1 0 1            0      0 1 1 1 1 1 0 1 0 1 0 1 0 1 0 0
0 1 1 1 1 0 1 0 1 0 1 0 1 0 0 0            0      0 1 1 1 1 0 1 0 1 0 1 0 1 0 0 0
0 1 0 1 0 0 0 1 0 1 0 1 0 1 0 0            1      0 1 0 1 0 0 0 1 0 1 0 1 0 1 0 1
0 0 0 0 0 0 0 0 0 1 0 1 0 1 0 0            0      0 0 0 0 0 0 0 0 0 1 0 1 0 1 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0            1      1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1
0 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0            0      0 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1 0 0 1 0 0 0 1 0            1      0 1 0 1 0 1 0 1 0 0 1 0 0 0 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1            0      1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 1            1      0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 1
0 1 0 1 0 1 0 1 0 0 0 1 0 1 0 1            1      0 1 0 1 0 1 0 1 0 0 0 1 0 1 0 1
0 1 0 1 0 1 1 1 1 1 1 1 1 0 0 1            0      0 1 0 1 0 1 1 1 1 1 1 1 1 0 0 0
0 1 1 1 1 0 1 0 1 0 1 0 1 0 1 0            0      0 1 1 1 1 0 1 0 1 0 1 0 1 0 1 0
0 0 1 0 0 1 0 1 0 1 0 0 1 0 1 0            1      0 0 1 0 0 1 0 1 0 1 0 0 1 0 1 1
                              ↑
                         LSB column
```

Fig.3

The application decoding the cover reads the eight Least Significant Bits of those bytes to recreate the hidden byte—that is 0110001—the letter "a." As you may realize, using this technique lets you hide a byte every eight bytes of the cover. Note that there is a fifty percent chance that the bit you're replacing is the same as its replacement, in other words, half the time, the bit doesn't change, which helps to minimize quality degradation. Fig.3 illustrates how the message 'HEY' is encoded in a 16-bit CD quality sample using the LSB method. Here the secret information is ‘HEY’ and the cover file is audio file. HEY is to be embedded inside the audio file. First the secret information ‘HEY’ and the audio file are converted into bit stream. The least significant column of the audio file is replaced by the bit stream of secret information ‘HEY’. The resulting file after embedding secret information ‘HEY’ is called Stego-file.

3.2. Message Encryption using playfair cipher

The Playfair cipher uses a 5 by 5 table containing a key word or phrase. Memorization of the keyword and 4 simple rules was all that was required to create the 5 by 5 table and use the cipher. To generate the key table, one would first fill in the spaces in the table with the letters of the keyword (dropping any duplicate letters), then fill the remaining spaces with the rest of the letters of the alphabet in order (usually omitting "Q" to reduce the alphabet to fit; other versions put both "I" and "J" in the same space). The key can be written in the top rows of the table, from left to right, or in some other pattern, such as a spiral beginning in the upper-left-hand corner and ending in the centre. The keyword together with the conventions for filling in the 5 by 5 table constitutes the cipher key.



W	A	T	C	H	KEY
B	D	E	F	G	
IJ	K	L	M	N	
O	P	Q	R	S	
U	V	X	Y	Z	

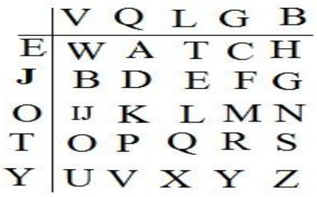
Fig. 4

Mod % 5
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
Z Y X W **V** U T S R **Q** P O N M **L** K J I H **G** F E D C **B** A

X = [E J O T Y]

Y = [V Q L G B]

All character are considered (both in Ascending and descending) which are divisible by mod 5 are stored in an array x & y based on the position of letter in fig. 4 the alphabetic position is considered in x & y. this can be visualise as shown below.



	V	Q	L	G	B
E	W	A	T	C	H
J	B	D	E	F	G
O	IJ	K	L	M	N
T	O	P	Q	R	S
Y	U	V	X	Y	Z

Fig 5

The encrypted message usually occurs in pairs and the length of the encrypted message is twice the length of the original message. E.g. Consider the message “JOB” which has been encrypted each letter is searched in a table and its corresponding row and column values are retrieved.

JOB = OV TQ JV

3.3. Conversion of encrypted message an 512 base character set

The encrypted message is converted into 512 base character set. ASCII characters can be split into the following sections.

0 – 31 Control codes
32-127 Standard, implementation-independent characters.
128-255 Special symbols, international character-generally, non standards characters.
512 base-character set

Control Codes: ASCII Characters 0 – 31

The following table lists and describes the first 32 ASCII characters, often referred to as control codes. The columns show the decimal and hexadecimal ASCII values for each code along with their abbreviated and full names. Descriptions are given to those most in use today.

Decimal	Code	Description
000	NUL	Null
001	SOH	Start Of Heading
002	STX	Start of TeXt
003	ETX	End of TeXt
004	EOT	End Of Transmission
005	ENQ	ENQuiry
006	ACK	ACKnowledge.
007	BEL	BELl. Caused teletype machines to ring a bell. Causes a beep in many common terminals and terminal emulation programs.
Decimal	**Code**	**Description**
008	BS	Backspace. Moves the cursor

		move backwards (left) one space.
009	HT	Horizontal Tab. Moves the cursor right to the next tab stop. The spacing of tab stops is dependent on the output device, but is often either 8 or 10 characters wide.
010	LF	Line Feed. Moves the cursor to a new line. On Unix systems, moves to a new line AND all the way to the left.
011	VT	Vertical Tab.
012	FF	Form Feed. Advances paper to the top of the next page (if the output device is a printer).
013	CR	Carriage Return. Moves the cursor all the way to the left,
014	SO	Shift Out
015	SI	Shift In
016	DLF	Data Link Escape
017	DC1	Device Control 1
018	DC2	Device Control 2
019	DC3	Device Control 3
020	DC4	Device Control 4
021	NAK	Negative AcKnowledge
022	SYN	SYNchronous idle
023	ETB	End of Transmission Block
024	CAN	CANcel
025	EM	End of Medium
026	SUB	SUBstitute
027	ESC	ESCape
028	FS	File Separator
029	GS	Group Separator
030	RS	Record Separator
031	US	Unit Separator

The Standard ASCII Characters: 32 – 127

ASCII Characters 32 - 127 are the standard, implementation-independent alphanumeric characters we work with every day. The tables below show the characters along with both their decimal and hexadecimal ASCII values.

Characters 32 - 64

The first table, which contains characters 32 - 64, contains the majority of the standard symbolic characters and the numbers from zero to nine.

Decimal	**Character**	**Decimal**	**Character**
032	Space	048	0
033	!	049	1
034	"	050	2
035	#	051	3
036	$	052	4
037	%	053	5
038	&	054	6
039	'	055	7
040	(	056	8
041	)	057	9
042	*	058	:
043	+	059	;
044	,	060	<
045	-	061	=
046	.	062	>
047	/	063	?
		064	@

Characters 65 – 127.

The second table, which contains characters 65 - 127, contains the standard Latin alphabet characters both lower and upper case, separated only by a few characters at 91 - 96 and 123 – 127.

Decimal	Character	Decimal	Character
065	A	078	N
066	B	079	O
067	C	080	P
068	D	081	Q
069	E	082	R
070	F	083	S
071	G	084	T
072	H	085	U
073	I	086	V
074	J	087	W
075	K	088	X
076	L	089	Y
077	M	090	Z

Decimal	**Character**	**Decimal**	**Character**
091	[	102	f
092	\	103	g
093	]	104	h
094	^	105	i
095	`_	106	j
096		107	k
097	a	108	l
098	b	109	m
099	c	110	n
100	d	111	o
101	e	112	p

Decimal	**Character**	**Decimal**	**Character**

113	q	121	y
114	r	122	z
115	s	123	{
116	t	124	|
117	u	125	}
118	v	126	~
119	w	127	Delete
120	x		

The Non-Standard ASCII Characters: 128 – 255.
The second half of the ASCII table holds the non-standard extension set of characters which may vary depending which computer system you may be using. One common – but in no way definitive –example of this extended set is as follows.

Characters 128 – 191

This first table contains characters 128 - 191, abstract symbols that appear in text from time to time.

Decimal	Character	Decimal	Character
128	·	157	Ÿ
129	.	158	¡
130	‚	159	¢
131	*ƒ*	160	Non -breaking space
132	„		
133	…		
134	†	161	¡
135	‡		
136	ˆ	162	¢
137	‰	163	£
138	Š	164	¤
139	‹	165	¥
140	Œ	166	¦
141	·	167	§
142	Ž	168	¨
143	·	169	©
144	·	170	ª
145	'	171	«
146	'	172	¬
147	"	173	-
148	·	174	®
149	·	175	¯
150	–	176	°
151	—	177	±
152	˜	178	²
153	™	179	³
154	š	180	´
155	·	181	µ
156	Ÿ	182	¶

Decimal	Character	Decimal	Character
183	·	188	¼
184	¸	189	½
185	¹	190	¾
186	º	191	¿
187	»		

Decimal	Character	Decimal	Character
192	À	224	à
193	Á	225	á
194	Â	226	â
195	Ã	227	ã
196	Ä	228	ä
197	Å	229	å
198	Æ	230	æ
199	Ç	231	ç
200	È	232	è
201	É	233	é
202	Ê	234	ê
203	Ë	235	ë
204	Ì	236	ì
205	Í	237	í
206	Î	238	î
207	Ï	239	ï
208	Ð	240	ð
209	Ñ	241	ñ
210	Ò	242	ò
211	Ó	243	ó
212	Ô	244	ô
213	Ö	245	õ
214	Ö	246	ö
215	×	247	÷
216	Ø	248	ø
217	Ù	249	ù
218	Ú	250	ú
219	Û	251	û
220	Ü	252	ü
221	Ý	253	ý
222	Þ	254	þ
223	ß	255	ÿ

The Non-Standard ASCII Characters: 256 – 512

This is based on Latin, Greek, Greek and Coptic, Extended Greek, Extended Latin, Cyrillic.
Characters 256-512.

Decimal	Character	Decimal	Character
256	.	309	ĵ
257	Ā	310	Ķ
258	ā	311	ķ

259	Ă	312	ĸ
260	ă	313	Ĺ
261	Ą	314	ļ
262	ą	315	Ļ
263	Ć	316	Ľ
264	ć	317	ľ
265	Ĉ	318	Ŀ
266	ĉ	319	ŀ
267	Ċ	320	Ł
268	ċ	321	ł
269	Č	322	Ń
270	č	323	ń
271	Ď	324	Ņ
272	ď	325	ņ
273	Đ	326	Ň
274	đ	327	ň
275	Ē	328	ŉ
276	ē	329	Ŋ
277	Ĕ	330	Ŋ
278	ĕ	331	Ō
279	Ė	332	ō
280	ė	333	Ŏ
281	Ę	334	Ŏ
282	ę	335	Ő
283	Ě	336	ő
284	ě	337	Œ
285	Ĝ	338	œ
286	ĝ	339	Ŕ
287	Ğ	340	ŕ
288	ğ	341	Ŗ
289	Ġ	342	ŗ
290	ġ	343	Ř
291	Ģ	344	ř
292	Ģ	345	Ś
293	Ħ	346	ś
294	ĥ	347	Ŝ
295	Ĩ	348	ŝ
296	ĩ	349	Ş
297	Ī	350	ş
298	ī	351	Š
299	Ĭ	352	š
300	ĭ	353	Ţ
302	Į	354	ţ
303	į	355	Ť
304	İ	356	ť
305	ı	357	Ŧ
306	Ĳ	358	ŧ
307	ĳ	359	Ũ
308	ĵ	360	ũ

Decimal	**Character**	**Decimal**	**Character**
361	Ū	413	Ί
362	ū	414	Ό
363	Ŭ	415	Ύ
364	ŭ	416	Ώ

365	Ů	417	Ϊ
366	ů	418	Γ
367	Ű	419	Θ
368	ű	420	Λ
369	Ų	421	Ξ
370	ų	422	Π
371	Ŵ	423	Σ
372	ŵ	424	Φ
373	Ŷ	425	Ψ
374	ŷ	426	Ω
375	Ź	427	α
376	ź	428	β
377	Ż	429	γ
378	ż	430	δ
379	Ž	431	ζ
380	Ž	432	η
381	ſ	433	θ
382	Ə	434	λ
383	Ơ	435	μ
384	Ơ	436	ξ
385	Ư	437	π
386	ư	438	ρ
387	Ǎ	439	ς
388	Ȧ	440	σ
389	Ǣ	441	τ
390	ǽ	442	φ
391	Ǿ	443	χ
392	ǿ	444	ψ
393	Ǔ	445	ω
394	ǔ	446	ώ
395	Ǖ	447	Ђ
396	ǖ	448	Љ
397	Ǘ	449	Њ
398	ǘ	450	Џ
399	Ǚ	451	Б
400	ǚ	452	Д
401	Ǜ	453	Ж
402	ǚ	454	З
403	Ǻ	455	И
404	ǻ	456	Й
405	Ǽ	457	Л
406	ǽ	458	Ц
407	Ǿ	459	Ч
408	ǿ	460	Ш
409	Ǻ	461	Щ
410	.	462	Ъ
411	Έ	463	Ю
412	Ή	464	Я

Decimal	**Character**	**Decimal**	**Character**
465	б	489	қ
466	д	490	Ҟ
467	ж	491	к
468	з	492	Ң
469	и	493	Ұ

470	й	494	Ұ
471	л	495	Ҳ
472	ц	496	ҳ
473	ч	497	Ҹ
474	ш	498	ҹ
475	щ	499	ә
476	ъ	500	ө
477	э	501	₣
478	я	502	₤
479	љ	503	№
480	њ	504	₫
481	ħ	505	₪
482	Ґ	506	‼
483	Ґ	507	•
484	Ғ	508	Ӟ
485	Ғ	509	ӟ
486	Җ	510	Ӿ
487	Җ	511	ӿ
488	Қ	512	·

E.g. Encryption using Playfair Cipher to encrypt i.e., consider the encrypted message “OV TV JV”, the corresponding message based on 512 base character set is 318,362,299,149,22

Consider a Message “Ŀ ūį •SYN”

ASCII Value of Message is Convert the decimal value to Binary Value.
Values are

100111110

101101010

100101011

010010101

000010110

2 bit from LSB position of each pixel is modified by the binary values of encrypted message which results in Stegano Image. This Stegano image is added to the audio file as Back Cover. The Front cover of audio file is just image of 640 * 480 of Black Pixel. When the Audio File is played only the Black background image or Front Cover is displayed, but image is hidden in Back Cover.

4. Conclusion

In this paper, we have devoted a methodology of Embedding message in a audio file. In a second manner Image steganography is used for the transportation of high level or top secret documents between international governments. Also it allows for copyright protection on digital files using the message as a digital watermark. Image steganography has many legitimate uses as it can be used by hackers to send viruses and Trojans to compromise machines. Ensuring data security is a big challenge for computer users. In this paper, the proposed method for embedding message in image, and in turn the image is embedded in audio file as cover media. The Algorithm will not work efficiently; if keywords contain repeated alphabets. A meaningful keyword may contain at most 17 distinct characters. Variation for the algorithm can be discovered, so it works with repeated character.

6. REFERNCES

[1]. Dennie Van Tassel, “Cryptographic techniques for computers: Substitution methods”, Vol.6.pp.241-249, Pergamon press 1970, Britain.

[2]. “An Improved Playfair Cipher Cryptographic Substitution Algorithm” Volume 2, No. 1, Jan-Feb 2011 International Journal of Advanced Research in Computer Science pp. 211-214

[3]. “An Enhanced and Secure Playfair Cipher by Introducing the Ferquency of Letters in any Plain text”, Journal of Current Computer Science and Technology, 1 (3), June 2011, 10-16.

[4] Medeni, M.B.O., Rabat, Morocco Souidi, E.-M “A novel steganographic method for gray-level images with four-pixel differencing and LSB substitution” Multimedia Computing and Systems (ICMCS), 2011

[5] Subba Rao Y.V, Brahmananda Rao S.S, Rukma Rekha “Secure Image Steganography Based on Randomized Sequence of Cipher Bits” , Information Technology: New Generations (ITNG), 2011

[6] R. Popa, An Analysis of Steganographic Techniques, The "Politehnica" University of Timisoara, Faculty of Automatics and Computers, Department of Computer Science and Software Engineering,1998

[7] Hide & Seek: An Introduction to Steganography: Niels Provos and Peter Honeyman, IEEE Security & Privacy Magazine, May/June 2003.

[8] Souvik Bhattacharyya, Indradip Banerjee and Gautan Sanyal,”A Survey of Steganography and Steganalysis Technique in Image, Text, Audio and Video as Cover Carrier”, in Journal of Global Research in Computer Science (JGRCS) VOL 2, NO 4 (2011),APRIL-2011.

[9] Manoj T H, Vimalanathan P, A Santha Rubia, sriVidya R "Secured Way of Encrypted Message Transmission using Audio File", IJCTA, Vol 3 (4), 1463-1466, Aug-2012.

[10] Domenico Bloisi and Luca Iocchi, "Image Based Steganography and Cryptography", Sapienza University of Rome, Italy.