AUSTRALIAN JOURNAL OF BASIC AND APPLIED SCIENCES

ISSN:1991-8178 EISSN: 2309-8414
Journal home page: www.ajbasweb.com

Text Cryptography Techniques Using Initial Vector and with/without keys.

[1]Tushar Chindalia and [2]Dr. Deepa V Jose

[1]Student, Christ University, Department of Computer Science, Christ University, Hosur Road, Bengaluru-560029, Karnataka, India.
[2]Assistant Professor, Department of Computer Science, Christ University, Hosur Road, Bengaluru-560029, Karnataka, India.

Address For Correspondence:
Tushar Chindalia, Christ University, Department of Computer Science, Dr. Deepa V Jose, Hosur Road, Bengaluru- 560029, Karnataka, India.
E-mail: chindalia.tushar@gmail.com

ARTICLE INFO

Article history:
Received 19 September 2016
Accepted 10 December 2016
Published 31 December 2016

Keywords:
Text Encryption, Text Decryption, Keyless, Plaintext, Cipher text, Text Cryptography, Symmetric Key.

ABSTRACT

Background: Cryptography is a technique of securing data for multiple purposes. This process includes both the process of encryption and decryption. Encryption is when the given data is covered from an understandable form to an un-understandable form, and decryption is the total opposite. It makes the un-understandable data to an understandable form.
Objective: This paper speaks about two cryptography methods, which focus on text cryptography. The idea is not only to encrypt text for secure communication, but also to do so quickly. This leads to the first proposed method, which completes the process without the use of a secret key. The encryption function uses the numeric values of the plaintext, from an initial vector and manipulates the plaintext to generate the cipher text. The next one is a keyed algorithm, which uses the key only to ensure the message is being decrypted to the authorized person. The key is of the alphanumeric form and it can be of any size.
Results: The implementation of the proposed algorithm resulted in a favorable outcome. The implementation has used the ASCII Table as the reference to populate the Initial Vector, and the max value was 256.
Conclusion: This paper proposes 2 algorithms which can provide sufficient security against the brute-force attack. The main application of such algorithms could be in storing sensitive data or sending secure e-mails. The strength of this algorithm is that it is much quicker and system resource saving, as there are no loops or repetitive steps followed.

INTRODUCTION

Crypto means “Concealed” or “Secret”, and Graphy means “A Descriptive Science” or “The Study of”. Together, Cryptography means “The Study of Concealing data”. Cryptography consists of 2 major parts, they are: Encryption and Decryption. Encryption is the process of generating a cipher text or a secret message, whereas Decryption is the opposite, that is; converting the cipher text to plaintext.

Cryptography is used for the following reasons: 1. Authentication, 2. Privacy, 3. Integrity, 4. Non-Repudiation and 5. Access Control.(A. Mathur, 2012)

There are 2 types of Recoverable Cryptography methods. They are: Symmetric-Key Cryptography, and Asymmetric-Key Cryptography (Also called Public-Key Cryptography). In Symmetric Cryptography, the key used to encrypt and decrypt the given data, is the same key, but in Asymmetric Cryptography, there is one key to encrypt the data and another key to decrypt the encrypted data.

This paper introduces 2 methods of cryptography, which are (1) Keyless Cryptography method, (2) Symmetric Key Cryptography method (referred as Keyed Cryptography, in this paper). These techniques are

To Cite This Article: Tushar Chindalia and Dr. Deepa V Jose., Text Cryptography Techniques Using Initial Vector and with/without keys.. ***Aust. J. Basic & Appl. Sci.,*** *10(18): 181-187, 2016*

developed to complete the process of encryption quickly and to withstand brute force attack. The actual cryptography method is the key of the algorithm, as in this is what makes the keyless cryptography secure, to a certain extend.

Literature Review:

In (A. Mathur, 2012), the author has proposed an algorithm for data cryptography (particularly text encryption and decryption) by altering the ASCII values of the plain text, based on the key. The key is first manipulated and then used. Also, this algorithm uses Symmetric Keys for this process. The only problem in this paper is that the key provided needs to be of the same length as the message.

The (J Gitanjali, Dr.N.Jeyanthi, C.Ranichandra, M.Pounambal, 2014) uses matrix multiplication and palindrome numbers to create a cipher text. It does so by operating on the ASCII values using a key, which consists of Palindrome Numbers and Unique Alphanumeric ID, which again is converted to ASCII. The ID provides authentication over the network. (J Gitanjali, Dr.N.Jeyanthi, C.Ranichandra, M.Pounambal, 2014) also sends the data in a set of 3 keys, making the task of the crypt-analyst challenging. On the decrypting side, the inverse of the encoding matrix is used and the plain text is obtained.

The author of (A. K. Bairagi, 2011), has introduced a new cryptography algorithm. This algorithm takes the ASCII codes of the plain text and based on the Evenness or Oddness of the ASCII Value, it is manipulated. After the encryption process, the final ASCII code is replaced in the Least Significant Bit of the Image (pixel value)(A. K. Bairagi, 2011). This image is the carrier of the cipher text. The only drawback is that it is a very long process and thus the execution time is large.

The paper (A. Singh and U. Jauhari, 2012), has 2 Cryptography technique combined. It uses text cryptography along with steganography to securely transmit data over the network. It follows the principle of symmetric key encryption, which is it uses the same key for encryption and decryption. Here, the key is an integer of length 5. The process of encryption takes the ASCII of the data and then performs arithmetic operations on it, with the digits extracted from the key. The final ASCII values are then hidden into an image, and transmitted.

Problem Statement:

There are a number of cryptography algorithms, but they concentrate on the security of the content. Some cases the security isn't the first priority, but the speed is. A good example would be e-mail. Here the data needs to be encrypted and send it but most of the available algorithms are time consuming. The algorithm proposed here is to achieve speed, by reducing the security level, as compared to other algorithms.

Proposed Work:

The proposed work has taken a few points as requirements, they are:

The receiver of the cipher text is the authorized receiver

There is an initial vector (IV), which contains a numeric value to every character. The values will be taken at the time of setup of the application. These values are the ones to be used for the purpose of encryption and decryption.

The Values in IV can be as large as needed, but the numbers allocated to every character must be in a sequence. That is, no numeric value can be skipped; all values from 0 to the largest numeric value must be feed to the IV.

This algorithm needs at least 2 characters as plain text and 2 characters as key, if it is being used, to work.

(1) Keyless Cryptography method:

This method works by taking the ASCII values of two consecutive characters and encrypting the first ASCII based on the second ASCII value. The data it accepts is only of String Format.

The Algorithms are as follows:

Algorithm: keyless_encryption(plain text)

Step 1: Calculate the length of the plain text.

Step 2: Do until the entire plain text is encrypted

- Step 1: Extract the character to be encrypted and the next character. If it's the last character, then take it and the first character of the cipher text.
- Step 2: Find the corresponding numeric value of each character form IV.
- Step 3: Add the 2 numeric values and perform modulo operation on the sum, with the Largest NumericValue in IV after incrementing it by 1.
- Step 4: The returned Character is the encrypted character; so append it to the cipher text.

Step 3: return the cipher text

Algorithm: keyless_decryption(cipher text)
Step 1: Calculate the length of the cipher text.
Step 2: Do until the entire cipher text is decrypted
  Step 1: Start from the last position. Extract the current character and the next character. If the last character needs to be extracted then extract the first character and pass this as the next character. Find their numeric values from IV. Perform subtraction, and modulo it with the largest value, after incrementing it by 1.
  Step 2: The value obtained needs to be fetched from the IV, to get its corresponding character.
  Step 3: The returned Character is the decrypted character; so append it to the plain text
Step 3: return the plain text

(2) Keyed Cryptography method:

This method works in the same manner as the keyless cryptography method, but instead uses a key to secure the encryption. The key and the data are to be in String Format.

The Algorithms are as follows:
Algorithm: key_encryption(plain text, key)
Step 1: Calculate the length of plain text.
Step 2: Calculate the length of Key.
Step 3: Encrypt the key, using the algorithm (keyless_encryption)
Step 4: Generate Special ASCII from Key
Step 5: Do until the plain text is encrypted
  Step 1: Extract the character to be encrypted and the next character
  Step 2: Call the encryption function, passing the 2 parameters (pass special ASCII along with last character, for encrypting last character)
  Step 3: The returned Character is the encrypted character; so append it to the cipher text.
Step 6: Append the Encrypted Key and Cipher Text into the cipher text
Step 7: Return the cipher text

Algorithm: key_decryption(cipher text, key)
Step 1: Calculate the length of cipher text.
Step 2: Calculate the length of Key.
Step 3: Encrypt the key, using the algorithm (keyless_encryption)
Step 4: Extract key from cipher text
Step 5: Check if the extracted key and the Encrypted Key match, if yes then continue. Otherwise report and exit.
Step 6: Generate Special ASCII from Key
Step 7: Do until the cipher text is decrypted
  Step 1: Extract the character to be encrypted and the next character
  Step 2: Call the encryption function, passing the 2 parameters (pass special ASCII along with last character, for decrypting last character)
  Step 3: The returned Character is the decrypted character; so append it to the plain text
Step 8: Return the plain text

Experimental Results:
Initial Vector:

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Fig. 1: Initial Vector values, part-1.

128	Ç	144	É	160	á	176	▒	192	└	208	╨	224	α	240	≡			
129	ü	145	æ	161	í	177	▒	193	┴	209	╤	225	ß	241	±			
130	é	146	Æ	162	ó	178	▓	194	┬	210	╥	226	Γ	242	≥			
131	â	147	ô	163	ú	179	|	195	├	211	╙	227	π	243	≤			
132	ä	148	ö	164	ñ	180	┤	196	—	212	╘	228	Σ	244	⌠			
133	à	149	ò	165	Ñ	181	╡	197	┼	213	╒	229	σ	245	⌡			
134	å	150	û	166	ª	182	╢	198	╞	214	╓	230	µ	246	÷			
135	ç	151	ù	167	º	183	╖	199	╟	215	╫	231	τ	247	≈			
136	ê	152	ÿ	168	¿	184	╕	200	╚	216	╪	232	Φ	248	°			
137	ë	153	Ö	169	⌐	185	╣	201	╔	217	┘	233	Θ	249	∙			
138	è	154	Ü	170	¬	186	║	202	╩	218	┌	234	Ω	250	·			
139	ï	155	¢	171	½	187	╗	203	╦	219	█	235	δ	251	√			
140	î	156	£	172	¼	188	╝	204	╠	220	▄	236	∞	252	n			
141	ì	157	¥	173	¡	189	╜	205	═	221	▌	237	φ	253	2			
142	Ä	158	₧	174	«	190	╛	206	╬	222	▐	238	ε	254	■			
143	Å	159	ƒ	175	»	191	┐	207	╧	223	▀	239	∩	255				

Fig. 2: Initial Vector values, part-2.

Implementation:

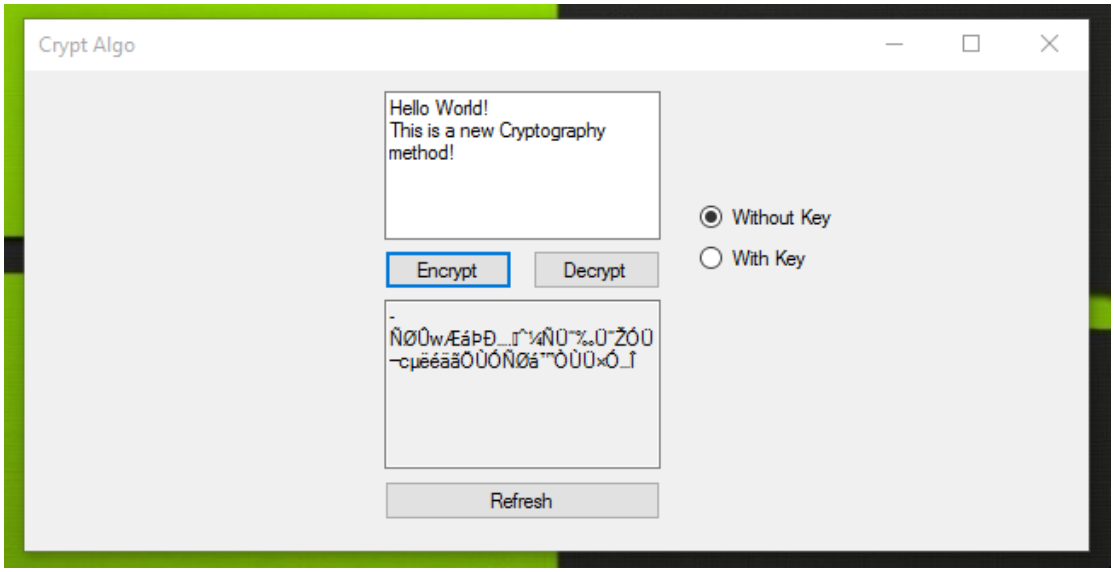

Fig. 3: Keyless Encryption using IV.

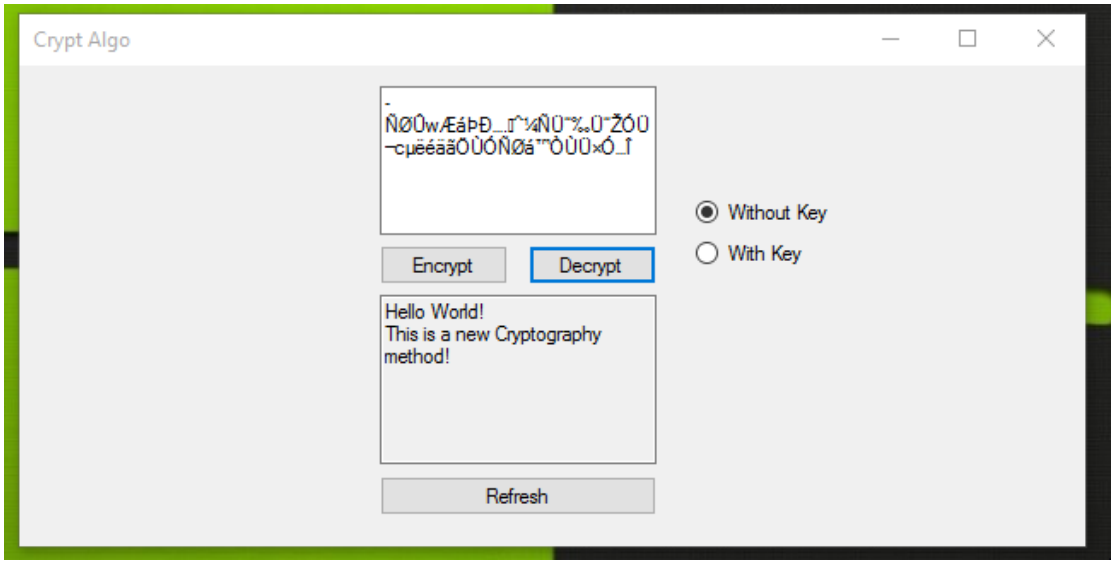

Fig. 4: Keyless Decryption using IV.

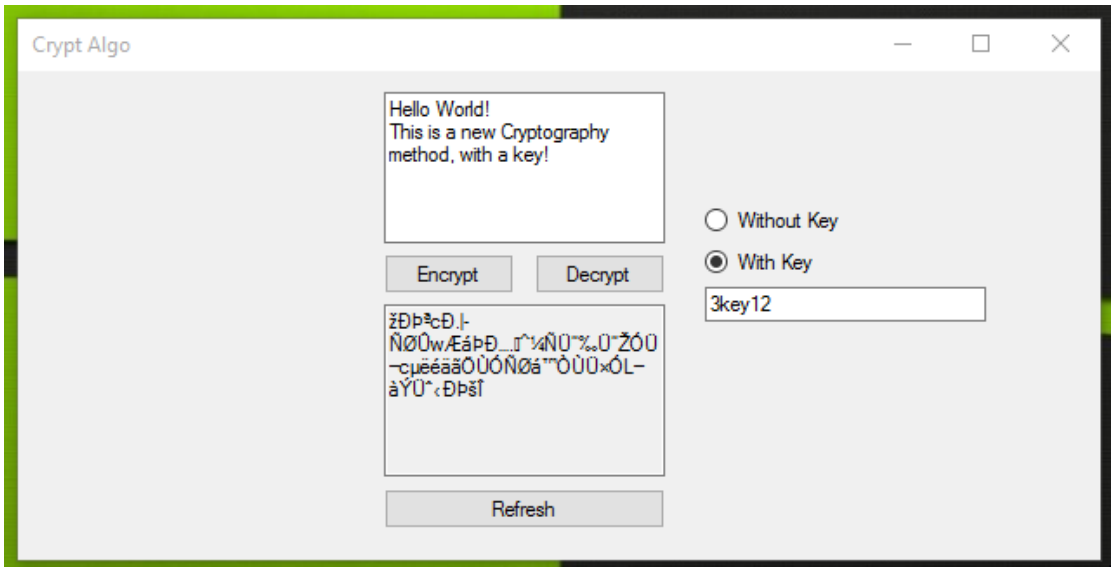

Fig. 5: Keyed Encryption using IV.

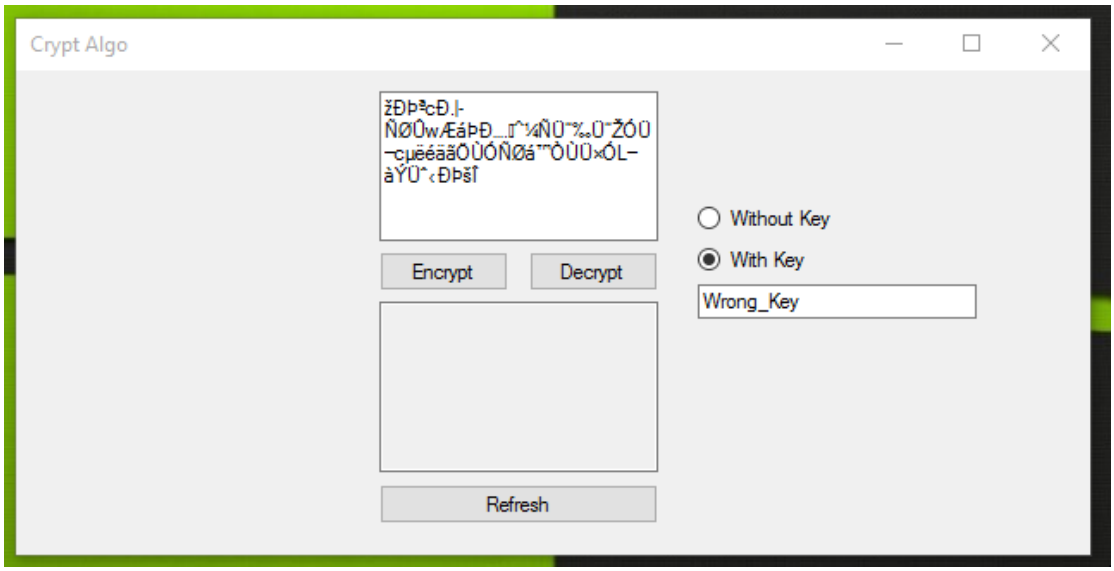

Fig. 6: Keyed Decryption using IV, and wrong Key.

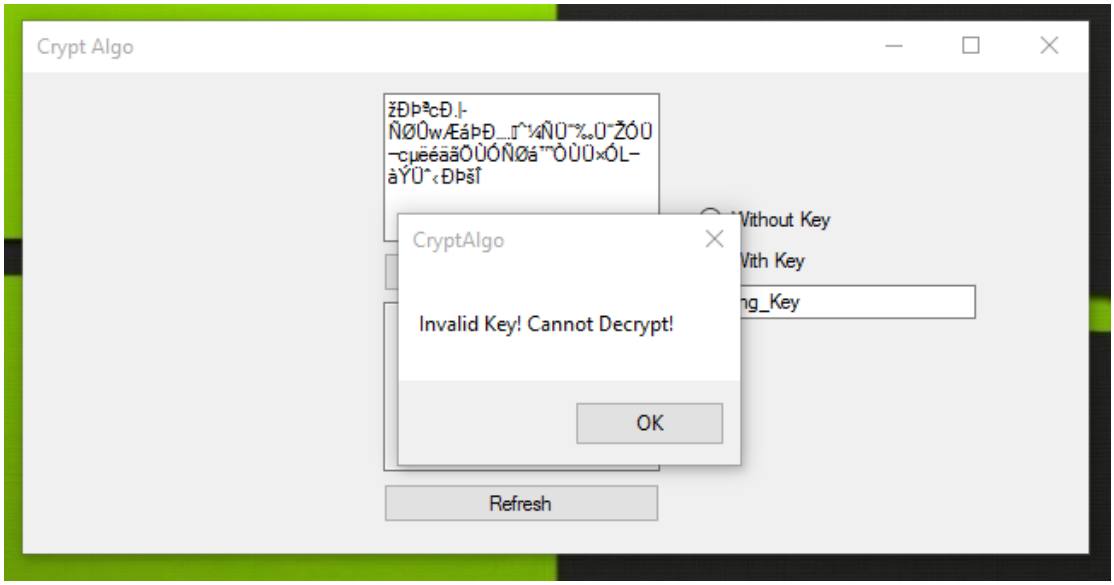

Fig. 7: Keyed Decryption using IV and wrong Key, Error.

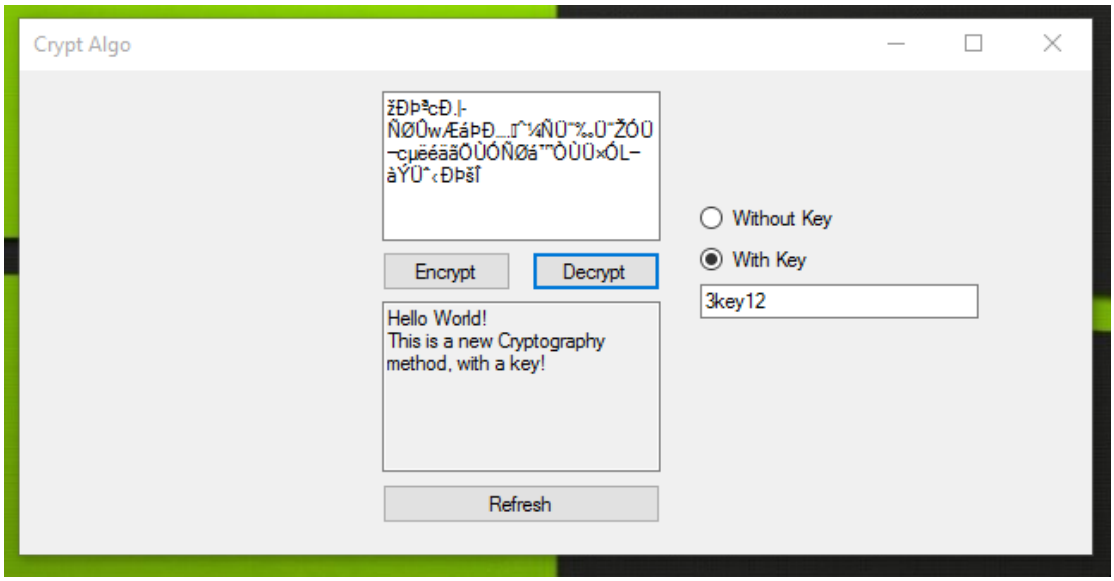

Fig. 8: Keyed Decryption using IV, and correct Key.

Mathematical Functionality:

The current algorithm gives the complexity depending on the IV. The larger the IV, the harder it will be to crack the encryption. The complexity is calculated in terms of factorial, as the number of possibilities depend on the size of the IV.

In the example the size of the IV was 256. So there are 256! possibilities of the IV, and 256! is:

857817775342842654119082271681232625157781520279485619859655650377269452553147589377440291360451408450375885342336584306157196834693696475322289288497426025679637332563368786442675207626794560187968867971521143307702077526646451464709187326100832876325702818980773671781454170250523018608495319068138257481070252817559459476987034665712738139286205234756808218860701203611083152093501947437109101726968262861606263662435022840944191408424615936000000000000000000000000000000000000000000000000000000000000000000000000. To find the correct combination would require a brute-force attack. This is the possibility of a keyless cryptography algorithm. In the keyed algorithm, the password would be required to start the decryption process, and the key can be of any size, containing alpha-numeric characters.

Conclusion:

The methods proposed in this paper are used to achieve a certain level of security with or without the use of secret keys. The process in the algorithm is a form of a key which makes cracking it, by using brute-force attack, very difficult, if not impossible. The application of this algorithm would likely be in mails for transferring

sensitive content, to intended users only, or storing sensitive or secretive data, or in any field where the data needs to encrypted and decrypted quickly.

Future Scope:

In the future, the keyed cryptography technique needs to be made much stronger, not only to brute-force attack but also other forms of attacks, and try to remove any existing faults, if found, in the current proposed method. Also the work needs to be optimized for more real-time applications like device communications, or focus it down to slower performing devices, so that the algorithm can perform quickly and keep resources available to other tasks.

REFERENCES

Joshi, A., 2011. "A Randomized Approach for Cryptography," City, pp: 293-296.

Bairagi, A.K., 2011. "ASCII based Even-Odd Cryptography with Gray code and Image Steganography : A dimension in Data Security," 01(02): 37-41.

Mathur, A., 2012. "A Research paper : An ASCII value based data encryption algorithm and its comparison with other symmetric data encryption algorithms," 4(09): 1650-1657.

Singh, A. and U. Jauhari, 2012. "Data Security by Preprocessing the Text with Secret Hiding," 3(3): 63-74.

Abhishek Anand, Abhishek Raj, Rashi Kohli and Dr. Vimal Bibhu, 2016. "Proposed Symmetric Key Cryptography Algorithm for Data Security", International Conference on Innovation and Challenges in Cyber Security, pp: 159-162.

Science, C. and S. Engineering, 2012. "Advance cryptography algorithm for improving data security, 1 1.," 2: 1.

Nilesh, D. and M. Nagle, 2014, "The new cryptography algorithm with high throughput," 2014 Int. Conf. Comput. Commun. Informatics Ushering Technol. Tomorrow, Today, ICCCI, pp: 3-7.

Bedrune, J.B. and F. Raynal, 2012. Cryptography: All-out attacks or how to attack cryptography without intensive cryptanalysis, 6: 3.

Gitanjali, J., Dr.N. Jeyanthi, C. Ranichandra, M. Pounambal, 2014. "ASCII Based Cryptography Using Unique lei, Matrix Multiplication and Palindrome Number.".

Uddin, M.P., M. Saha, S.J. Ferdousi, M.I. Afjal and M.A. Marjan, 2014. "Developing an efficient solution to information hiding through text steganography along with cryptography," 2014 9th Int. Forum Strateg. Technol. IFOST, pp: 14-17.

Pareek, N.K., 2005. "Cryptography using multiple one-dimensional chaotic maps," 10: 715-723.

Pareek, N.K., V. Patidar and K.K. Sud, 2003. "Discrete chaotic cryptography using external key," 309: 75-82.

Neha Tayal, Ritesh Bansal, Shailender Gupta and Sangeeta Dhall, 2016. "Analysis of Various Cryptography Techniques: A Survey", International Journal of Security and Its Applications, 10(8): 59-92.

Uddin, P., A. Marjan, N.B. Sadia and R. Islam, 2014. "Developing a Cryptographic Algorithm Based on ASCII Conversions and a Cyclic Mathematical Function," pp: 0-4.

Bassous, R., R. Bassous, H. Fu and Y. Zhu, 2015. "Ambiguous Multi-Symmetric Cryptography," *IEEE Int. Conf. Commun.*, pp: 7394-7399.

Pape, S. and N. Benamar, 2008. "Using identity-based public-key cryptography with images to preserve privacy," IFIP Int. Fed. Inf. Process., 262: 299-310.

Shankar, T.N., 2010, "Cryptography by Karatsuba Multiplier with ASCII Codes," 1(12): 53-60.

Singh, U. and U. Garg, 2013. "An ASCII value based text data encryption System," 3(11): 1-5.

Vismita Jain and Vijay Kumar Verma, 2016. "A Recent Study of Various Symmetric Key Based Cryptography Methods", International Journal od Science Technology Managemant and Research, 1(3): 14-20.

Asafe, Y., A.E. Edwin and O.F. Mercy, 2014. "Cryptography System for Online Communication Using Polyalphabetic Substitution Method," 2157: 2151-2157.