SRI RAMSHALAKA: A VEDIC METHOD OF TEXT ENCRYPTION AND DECRYPTION

Rajkishore Prasad
University Department of Electronic Science ,
B.R.A. Bihar University Muzaffarpur, Bihar, India.
Email:profrkishore@yahoo.com

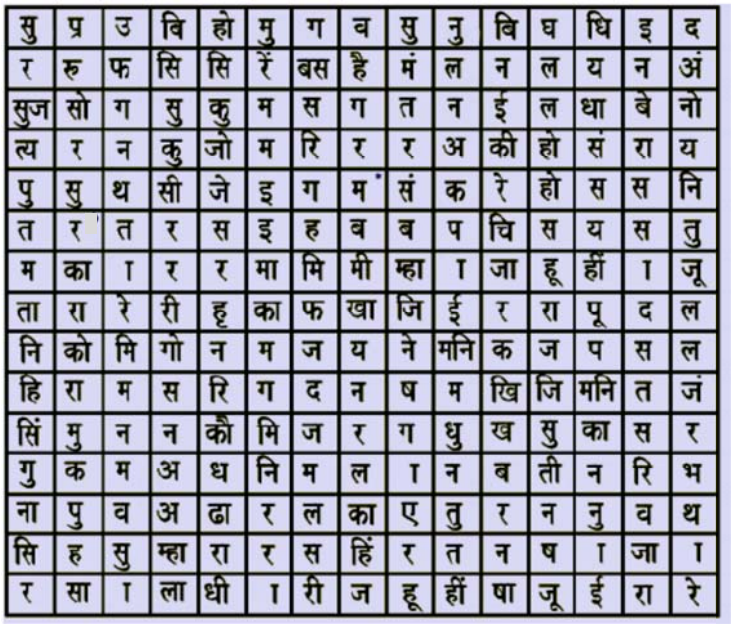
Abstract: This paper investigates the usability of *SriRamshalakha*, a vedic tool used in Indian Astrology, in the encryption and decryption of plain English text. *Sri Ram Shalaka* appears in *Sri RamChartmanas*, one of the very popular sacred epic of Hindu religion, written by great Saint Tulsidasji. *SriRamshalakha* is used to fetch/infer the approximate answer of questions/decisions by the believers. Basically, the said *shalaka* embed nine philosophical verses from *Sri RamCharitmanas* in a matrix form based on which answers to queries are inferred and ingrained. However, none of the verses are visible and directly readable. Here we take *SriRamshalakha* as the ancient Indian method of text encryption and decryption and based on the same algorithms for the encryption and decryption of plain English text are proposed. The developed algorithms are presented with examples and possibility of its use in steganography and text to image transformation are also discussed.

***Keywords*: Encryption; Decryption; cryptography; steganography.**

1. Introduction

SRI RamShalaka Prashanawali is not being introduced here rather it is well known to people who know *Sri RamCharitmanas* , a Hindi epic by great Saint Tulsidasji [Tulsidas, 1563]. *SRI RamShalaka Prashanawali* is one of the tools used in Vedic Astrology in prediction. It is used to answer or predict the outcome of any activity or plan in advance and thus it is like a magic or miracle for those who have faith in it. It is said that this prediction method had been used by *Sri Ram* to perform his day to day royal and personal duties. In the name *Sri RamShalaka Prashanawali* the word *Sri Ram* refers to Lord Ram, *Shalaka* means matrix and the word *prashanawali* means questionnaire. Originally, who developed this method first in the Ancient India is not known but it became part of the *Sri Ram CharitManas*, an epic written in 16th century by Tulsidas thus it is also known as Tulsidas' *SriRamShalaka Prashanawali* . Hereafter the word *shalaka* will alone refer to *SriRamShalaka*. *SriRamShalaka* as appears in the *SriRamCharitManasa* epic is shown in Fig.1.

Basically, *SriRamShalaka* hides nine verses or *chopai* of Philosophical meaning, from which solution to any actions/questions related to our activities and lives are inferred and ingrained, in a square matrix of size 15x15. All the nine verses used in *shalaka* are given in the first column of the Table 1. These verses are broken into independent *akshar* (Hindi Letters) and such *akshars* are entered as the elements of 15x15 matrix, as shown in Fig.1, following some rules to be discussed later. The constituent *akshars* (Hindi letters) used from each verse in the construction of *shalaka* are also shown in the third column of the Table.1. Each verse in *shalaka* has been represented by 25 unique *akshars or* Hindi letters despite the fact that each verse does not contain exactly 25 Hindi letters. For the same either some letters have been combined or separated to make total number of representative elemental *akshar* 25. In the *shalaka* shown in Fig.1, it is hard to see/notice presence of any of the verses in directly readable form, however, following certain process, to be discussed later, each of the nine verses can be obtained or decoded. This means actual text can be made concealed in *Shalaka* format and all the hidden verses can be decrypted using some key. In the *shalaka* the characters of the original text are not changed but their orders are changed. There have been developments of many encryption algorithms for the text by scrambling the characters in different ways. *SriRamShalaka* uses its own way based on matrix theory and to the best of my knowledge no work on use of *SriRamShalaka* in the text encryption has been reported so far. Thus in this paper *SriRamShalaka* is taken as one of the interesting ways of message encryption/decryption using the matrix theory and based on the same, algorithms are proposed to encrypt/decrypt plain English text.



सु	प्र	उ	वि	हो	मु	ग	व	सु	नु	वि	घ	धि	इ	द
र	रु	फ	सि	सि	रें	बस	है	मं	ल	न	ल	य	न	अं
सुज	सो	ग	सु	कु	म	स	ग	त	न	ई	ल	धा	बे	नो
त्य	र	न	कु	जो	म	रि	र	र	अ	की	हो	सं	रा	य
पु	सु	थ	सी	जे	इ	ग	म	सं	क	रे	हो	स	स	नि
त	र	त	र	स	इ	ह	ब	ब	प	वि	स	य	स	तु
म	का	ा	र	र	मा	मि	मी	म्हा	ा	जा	हू	हीं	ा	जू
ता	रा	रे	री	हृ	का	फ	खा	जि	ई	र	रा	पू	द	ल
नि	को	मि	गो	न	म	ज	य	ने	मनि	क	ज	प	स	ल
हि	रा	म	स	रि	ग	द	न	ष	म	खि	जि	मनि	त	जं
सिं	मु	न	न	कौ	मि	ज	र	ग	धु	ख	सु	का	स	र
गु	क	म	अ	ध	नि	म	ल	ा	न	ब	ती	न	रि	भ
ना	पु	व	अ	ढा	र	ल	का	ए	तु	र	न	नु	व	थ
सि	ह	सु	म्हा	रा	र	स	हिं	र	त	न	ष	ा	जा	ा
र	सा	ा	ला	धी	ा	री	ज	हू	हीं	षा	जू	ई	रा	रे

Fig,1 *SriRamshalaka Prashanawali* (Taken from *Sri RamCharitmanasa* written by Sant Tulsidasji in 16th Century)

Table 1. Verses used in *Sri RamShalaka* and their characterization

9 VERSES/COUPLET(*CHOPAI*)	**No. of Letters**	***25 AKSHARs* used in *Shalaka***
1. सुनु सिय सत्य असीस हमारी पूजिहि मन कामना तुम्हारी (Sun siya satya asis hamari, Pujahi mankaamna tumahri)	**23**	सु नु सि य स त्य अ सी स ह म ा री पू जि हि म न का म ना तु म्ह ा री
2. *प्रबिसि नगर कीजे सब काजा हृदयँ राखि कोसल पुर राजा* (*Prabasi nagar keeje sab kaaja, Hirde rakhi kausalpur raja*)	***24***	प्र बि सि न ग र की जे स ब का जा हृ दयँ रा खि को स ल पु र रा ज ा
3. *उधरहिं अंत न होइ निबाहू काल नेमि जिमि रावन राहू* (Udhre ant na hoi nibahu, kalnemi jimi ravan rahu)	***23***	उ ध रहिं अं त न हो इ नि ब ा हू का ल ने मि जि मि रा व न रा हू
4. *बिधि बस सुजन कुसंगत परहीं फनि मनि सम निज गुन अनुसरहीं* (*Udhre ant na hoi nibahu, kalnemi jimi ravan rahu)*	***29***	बि धि बस सुज न कु सं ग त प र हीं फ नि मनि स म निज गु न अ नु स र हीं
5. *होइहि सोइ जो राम रचि राखा कोकरि तर्क बढावै साखा* (*Hoi hai soi jo Ram rachi rakha, Ko kari tarak badavahi saka)*	***22***	हो इ हि सो इ जो रा म र चि रा खा को क रि त र्क ब ढ़ा वै सा खा
6. *मुद मंगलमय संत समाजू जो जग जंगम तीरथ राजू (Mud mangalmay sant samaju, jimi jag langam teerath raju)*	***23***	मु द मं ग ल म य सं त स मा जू जो ज ग जं ग म ती र थ रा जू
7. *गरल सुधा रिपु करहीं मिताई गोपद सिंधु अनल सितलाई (Garal sudha ripu karay mitai, Gopad sindhu anal sitlai)*	***25***	ग र ल सु धा रि पु क र हीं मि ता ई गो प द सिं धु अ न ल सि त ला ई
8. *बरून कुबेर सुरेस समीरा रन सन्मुख धरि काहुँ न धीरा (Varun kuber sures samira, ran sammukh dhar kah na dheera)*	***24***	ब रू न कु बे र सु रे स स मी रा र न स न्मु ख ध रि का हुँ न धी रा
9. *सुफल मनोरथ होहुँ तुम्हारे रामु लखनु सुनि भए सुखारे (Sufal manorath hoy tumhare, Ram Lakhan suni bhaye sakhare)*	***27***	सु फ ल म नो र थ हो हुँ तु म्हा रे रा मु ल ख नु सु नि भ ए सु ख ा रे

The sciences of Cryptography and Steganography [A.Joseph Raphael 2011] are well known and have very old origin. Using these techniques, information is manipulated to cipher or hide their appearance respectively. Cryptographic techniques scramble the original message prohibiting direct reading while Steganography [Sharma, January 2012] hides message in other media so that it cannot be read directly. Thus in the modern digital communication these two techniques are very important for secured and private communication for the transfer of secret messages. A few decades back cryptography was important only for the defense and government communication, but nowadays it is required in every aspect where we do exchange of information digitally such as in use of ATM, Credit Card, Smart Cards, electronic home systems, RFID tags etc [M. Savari, 2012]. Accordingly, many cryptographic algorithms have been developed for different sectors and are also available for the private uses. The account of chronological development of encryption method is interesting. The commence of use of encryption can be traced back into history from use of different methods of secret communications in different civilizations and cultures. There is evidence that around 4000 years back ancient Egyptians used hieroglyphic method for secret communication. Around 500-600 B.C. Hebrew scribes used ATBASH, a reverse alphabet simple solution cipher. Julius Caesar, from 50-60 B.C., used a simple substitution cipher for his government related communications [Srikant K., 2012]. The art of secrete writing and private communication in ancient India was also well developed and used in financial, administrative, religious and personal record-keeping and communications. All the important mathematical formulae were sonically encrypted in devotional hymns and lyrics. In the book *The Codebreakers* [Khann D, 1967] author has given brief description on cryptography in ancient India. In the *Kamsutra, Vatsayana* has described *mlecchita-viklapa* as cryptographic language and its knowledge for women as one of the required arts, essential to maintain secrecy of the personal relation and communication. There was development of two types of writing namely *Kautilayama* in which letters were substituted based on phonetic relation and other is *Muladivaya* in which paired letters were used as cryptographic unit [Kak S., 2006]. There had been development of another version of private messaging such as methods based on knowledge of dialects and messaging by wrist-finger gesture known as *akshar mustika kathanam* [Kalayanraman S.]. The other important ancient cryptographic technique in India was *Katayapadi* in which different consonants are mapped into digits from 0 to 9 and vowels are left free to use as per convenience in forming meaningful words [Kak S., 2006]. *Aryabhatta* used another very complex type of coding to encrypt astronomical figures into words [Kak S. 2006,1998]. Interestingly, in ancient India even birds (parrot/pigeons) had been used to carry secret messages. There are also descriptions of use of telepathic technology in the ancient Indian texts. Similarly, other methods of secret communication were used in different civilizations, however, the notion of modern cryptography, based on mathematical concept, took birth form the work of C.E Shanon in 1948 [Shannon C. , 1948] and became practical with the advent of microcomputers in 1960. The sound mathematical foundation and invention of digital computers led together development of different encryption/decryption techniques.

The purpose of encryption is to prohibit unauthorized reading and modification of the data. The plain text is modified according to some keys to produce what is called cipher/encrypted text. The cipher text is decrypted using same or different code/key to retrieve the original text. Cryptography system, based on key, can be broadly categorized into two classes namely symmetric key system and public key system. In the symmetric key system single key is used for encryption and decryption of the message while in the public key system, public key is used to encrypt the message and private key is used to decrypt the cipher text. The method of decoding cipher text is known as cryptanalysis and thus encryption should be strong enough to abort the efforts of attackers or any unauthorized cryptanalyst to decode the ciphered message/information. Currently, there are available many key based encryption algorithms such as DES, RSA, PGP, Elliptic curve, and others, however, use of the encryption started since antiquity.

In the most of the cryptographic systems [Stallings, 2005], encryption algorithms are based on the process of either substitution or transposition or combination of both. In the substitution based algorithms, the text element of plain text is mapped into another element by replacement of letters according to some rules. The replaced letter may be single, double or triplets or combination of other symbols. The elements of plain text retain their order in the cipher text but the text elements are changed. In the simple substitution encryption, known as monographic substitution, substitution takes place over single element, however, in polygraphic method substitution is done on group of letters. Encryption algorithm such as Hill Cipher, method used by Julius Caesar, etc. belong to this category. In the transposition based encryption algorithm elements of plain text are rearranged using some plan and thus each element of plain text retains its identity in the cipher text despite change in its position. Textual letters in such methods are scrambled based on geometrical patterns or along rows or columns and thus the relative position of each element is changed in the cipher text. The plain text is obtained by reversing the process of transposition so followed. Under these technical aspects, the method of text encryption used in the construction of *Sri RamShalaka* is based on transposition principle only.

The rest of this paper is organized as follows. In the next section text scrambling in *SriRamShalaka* is presented. Section third deals with development of encryption algorithms for the plain English text. In the fourth

section, examples of encryption and decryption using proposed algorithm is presented. The last section, which is followed by references, concludes the paper.

2. Text Encryption and Decryption in *SriRamShalaka*

In order to investigate usability and development of *SriRamShalaka* like encryption algorithm for plain English text some basic features of *shalaka* are essential to understand. This will expose how a transposition based method has been used in *shalaka* in the text encryption using matrix theory. Some of such basic features are listed below

1. *SriRamShalaka* is a square matrix of size 15x15 whose elements are *akshar* (Hindi letters) taken from nine *chopai* (couplets/verses) shown in first column of table 1. Each couplet has been broken down to the character/letter level as shown in column third of table 1.

2. The number of Hindi letters in each couplet is not same as shown in the table. However, all the nine verses have been entered in *shalaka* using 225 Hindi letters, securing 25 elements for each verse/couplet in the square matrix of size 15x15. As shown in the middle column of the Table.1, in some couplets the number of *akshar*/Hindi letters is less than 25 while in others it is more than 25 but each couplet has been given 25 places in the square matrix (*shalaka* of Fig.1) either by combining two letters or separating vowel and consonant parts of the letters. What has been the real plan for the selection of a particular letter for such combination or separation is neither depicted nor obvious. But by providing equal number of representative spaces (no of elements) to each couplet, the outcome of the *shalaka* has been made equi-probable for each involved verse/couplet.

3. The characterizations, selection of particular akshar of a couplet and placement in shalaka have been done systematically. It has been taken in group of nine taking one *akshar* at a time from each couplet i.e. firstly, first character/Hindi letter of each couplet has been taken then second character from each couplet and so on. The blocks of such nine characters have been placed along the rows of the 15x15 *shalaka* (matrix) of Fig.1 placing each representative *akshar* of a couplet as one element in continuation.

4. Each element of the 15x15 matrix is tagged uniquely to a couplet and each couplet is tagged to 25 unique places (elements) in the *shalaka*.

5. The *Shalaka* has been presented in square matrix format providing equal weightage to each verse (*chopai*) by making the outcome of reading of *shalaka* equi-probable, Probability of getting any *chopai/couplet* from the *shalaka* /matrix is 1/9.

6. Each *chopai* can be obtained from *shalaka* by concatenating together a *akshar /* character with its 9^{th} successors or predecessors in the same order following along the rows in continuation. The starting *akshar*/element decides which couplet will be decrypted. Palpably, in order to read *Sri Ram Shalaka* one needs to know the number of verses that has been encrypted. Thus this information is the key for decoding the hidden text.

Considering the above facts *SriRam Shalaka* can be formed in any shape and read accordingly. The rows and columns of constructed *shalaka* can be manipulated /modified using matrix theory to make more complex representations. There are many possibilities in this regard. The simple observation of the *shalaka* does not reveal any of the *chopai* legible. The text of *chopai* is hidden or encrypted. It can be read only if one knows the decoding method and key, the number of verses that has been encrypted, as described above. Thus *Sri Ram Shalaka* represents method of text encryption and decryption. The aim of this paper is to show usability of this Vedic method in the encryption and decryption of the plain English text.

3. *SriRamShalaka* based Algorithm for English Text

SriRamShalaka based algorithm for text encryption and decryption can be developed for a single sentences/message or multiple sentences. *SriRam Shalaka* encrypts nine ***chopai or verses*** in 15x15 matrix, however, its technique can be used to construct *shalaka* even for fewer verses. The key point of our concern is to do *Shalaka* like scrambling of the text. Here similar algorithms for text scrambling for single and multiple English sentences are proposed and presented. Algorithmic steps for both conditions are given below:

a. For Single sentence/message

1. Take the plain English text which is to be encrypted and count number of letters, say it's *L*.

2. Next segment the text into *N* separate parts/message of group of *M* letters such that the whole text becomes matrix of letters of size *NxM, known here as message matrix*. In Order to do such breaking one may need to add some extra characters (letters), say number of added extra character is *E*, such that $N*M=L+E$ where '*' denotes multiplication.

3. If one likes to make square shaped *shalaka of size NxN* then the number of extra characters added should be such that $N*N=L+E$.

4. After representing the given sentence into *NxM* (for rectangular *shalaka*) or *NxN* (for square *shalaka*) matrix , the *shalaka* is formed by placing each column of the message matrix along the rows in continuation until all the columns of matrix are finished. For square shaped message matrix this can be done by simply taking transpose of the matrix, however, for the *NxM* message matrix it is not so and required to be arranged for each column. The end of this gives representation of the original message in the form of *SriRamShalaka*.

5. Then the encrypted message can be generated from *shalaka* by arranging rows/column in continuation or using other secret plans/key.

6. By following the reverse process in decryption, message matrix from *shalaka* can be obtained. For the same one needs to reverse all the steps followed to construct *shalaka*.

b. For multiple sentences/messages

In the case of multiple messages /sentences, whole text or all sentences can be taken together, in continuation, as one message and *shalaka* representation can be obtained as outlined above. However, the original *Sri Ram Shalaka* itself hides nine separate verses (messages) and thus, in the same spirit, *shalaka* for the multiple plain English sentences can be made by following steps given below

1. Let *N* messages are to be represented in *shalaka*

2. Count the number of letters in each message/sentence and equalize the number of letters in each sentence/message by adding extra character in each sentence/message such that each message now consists of *L* letters. This makes whole text as *N*x*L* matrix which is called here message matrix.

3. Count the total number of characters in message matrix, obtained above in step 2. Here it is N*L.

4. To obtain rectangular *shalaka* each column of the message matrix is arranged along rows of *N*x*L* matrix in continuation. Here one may try to arrange rows/diagonals of message matrix along the columns of the *N*x*L* matrix to get the *shalaka* in different form.

5. In order to represent the message matrix in a square shaped *shalaka,* one needs to find a number P which is square root of *N*L*. If the total number of element *N*L* is not a perfect square, each row of the message matrix is padded with some extra character, say total number of padded character=*E*, such that (*N*L+E*) becomes a perfect square and square root of the same is taken as the dimension of the *shalaka* . Here let us take *P*x*P shalaka* which can accommodate all the elements *N*L* (may be more if some extra character has been added) providing equal number of entry spaces for each row of the message matrix. Now each column of the message matrix is required to be arranged in continuation along the rows of *P*x*P* matrix. This gives square shaped *shalaka* of message matrix. Here one may try to arrange rows/diagonal elements of the message matrix along the columns of *P*x*P* matrix to get different *shalaka*.

6. The encrypted message can be generated by arranging rows or columns of the *shalaka* or using some other secret plans/keys. The *shalaka* can also be manipulated differently to obtain encrypted message. The scope for manipulations of *shalaka* is numerous, depending upon application of matrix theory.

7. The encrypted message can be decrypted by reversing all the processes followed in encryption. The simple way to decrypt the message is to collect together all the *N*th letters, starting from any character of the *shalaka*.

These algorithms, as stated above, have been implemented in MATLAB. In the next section some examples of encryption/decryption are presented.

4. Examples of Encryption and Decryption using *Shalaka* based algorithms

In this section some examples of encryption and decryption of plain English text using *SriRamShalaka* based algorithms, as proposed above, are presented for each case.

Let us take a single message /sentence

message1='Heaven has been discovered in space on Jan 14, 2013',

to be encrypted using *shalaka* algorithm. The blanks or spaces are replaced by '@'. It is not essential to do such replacements of inter-word blank spaces. It can be removed but for simplicity it is being replaced by some special character '@'. The modified message becomes

A1= HEAVEN@iS@dISCOVERED@iN@sPACE@oN@jAN@14,2013.

In this modified message first character of words succeeding blank is in lower case which has been done to identify preceding blank space in decoding the message if blank spaces are removed. This act as tag to inter-word spacing when blank spaces are removed, however, not required here as blanks are replaced. Now let the modified message is arbitrarily segmented into *N=6* sub-messages and message matrix so obtained is shown in Fig.2. The total number of the character in original message is 51 which has been broken into 6x9 matrix called here as message matrix. The last row of the message matrix has been padded with extra character '#' to make total number of element 54 to complete 6x9 matrix. These extra elements can be placed anywhere, at

MESSAGE MATRIX

H E A V E N @ H A

S @ B E E N @ D I

S C O V E R E D @

I N @ S P A C E @

O N @ J A N @ 1 4

, 2 0 1 3 # # # #

Fig.2 Message matrix of size 6x9 for single message.

Text-Shalaka(Square shaped)

H S S I O , E @

C N N 2 A B O @

@ 0 V E V S J 1

E E E P A 3 N N

R A N # @ @ E C

@ # H D D E 1 #

A I @ @ 4 # # #

# # # # # # # #

Fig.3 Square shaped (8x8) *shalaka* for message matrix shown in Fig.2. The extra character is again inserted at the last.

Text-Shalaka(rectangular)

H S S I O , E @ C

N N 2 A B O @ @ 0

V E V S J 1 E E E

P A 3 N N R A N #

@ @ E C @ # H D D

E 1 # A I @ @ 4 #

Fig.4 Rectangular *Shalaka* of size 6x9 for the message matrix shown in Fig.2.

random, in any row of the message matrix, however, for the simplicity it has been placed as the last three elements of the last row of the message matrix. The total number of elements in the message matrix is 54 which can be transformed into square shaped *shalaka* or rectangular *shalaka* as shown in Fig.3 and Fig.4 respectively. In order to make square shaped *shalaka* with 54 elements of the message matrix, at least 10 extra characters are required to be added in the message matrix and resulting square shaped *shalaka* of size 8x8, constructed using above mentioned algorithm, is shown in Fig.3. The formation of rectangular *shalaka* does not require addition of any extra character in the message matrix as it is of size of the message matrix.

The square shaped or rectangular *shalaka* obtained so, represents encrypted message in the format of *SriRamShalaka from* which all the 6 messages (rows) of message matrix can be obtained by concatenating 6th succeeding or preceding characters, in the same order, starting from any character. However, each element of the *shalaka* is uniquely tagged to a particular row of the message matrix, the decoded message depends on the position of starting element/letter chosen for the same. In this way whole message matrix of Fig.2, can be obtained from where original messages can be discovered. Further complicated cipher text can be obtained by applying different matrix manipulation techniques on the *shalaka* matrices and original message can be decoded by reversing the process followed in encryption. Such manipulations are beyond the scope of this paper, however, encrypted messages *row_mes*, *col_mes*, and *diag_mes* taken along row , columns and along diagonal respectively of both *shalaka* are shown below

Encrypted message from Square shaped *shalaka* of Fig.3

row_mes
=HSSIO,E@CNN2ABO@@0VEVSJ1EEEPA3NNRAN#@@EC@#HDDE1#AI@@4##########

col_mes
=HC@ER@A#SN0EA#I#SNVENH@#I2EP#D@#OAVA@D4#,BS3@E##EOJNE1##@@1NC###

diag_mes
=#A#@I#R#@#EAH@#@END4#C0E#D##HNVP@E##SNEA@1#S2V3E#IASNCOBJN,O1E@@

Encrypted message obtained from rectangular *shalaka* of Fig.4

row_mes =HSSIO,E@CNN2ABO@@0VEVSJ1EEEPA3NNRAN#@@EC@#HDDE1#AI@@4#

col_mes =HNVP@ESNEA@1S2V3E#IASNCAOBJN@I,O1R#@E@EAH@@@END4C0E#D#

diag_mes =E@1P@#VAEANE3CIHNVN@@S2SN#@SAJRH4IB1AD#OOEND,@E#E@E@0C

As stated above once the message is represented in matrix form, the resulting *shalaka* can be modified to make decryption cumbersome by attackers. One of such modifications by rearranging element of square shaped *shalaka* along the diagonal, any diagonal direction can be chosen for this process, is shown below in Fig.5. The encrypted messages *row_mes*, *col_mes*, and *diag_mes* taken along row , columns and along diagonal respectively of modified *shalaka* are also shown. The scopes for such manipulations, based on matrix theory, in *Shalaka* represented text are numerous.

Modified Shalaka

H C S @ N S E 0
N I R E V 2 O @
A E E A , A # N
P V B E # I H #
A S O @ # @ D @
3 J @ # @ D @ N
1 # 4 E E N # #
1 C # # # # # #

row_mes =HCS@NSE0NIREV2O@AEEA,A#NPVBE#IH#ASO@#
@D@3J@#@D@ N1#4EEN##1C######

col_mes = HNAPA311CIEVSJ#CSREBO@4#@EAE@#E#NV,##
@E#S2AI@DN#EO#HD@##0@N#@N##

diag_mes =11C3##AJ4#PS@E#AVO#E#NEB@@N#HIEE#
D##CRA#@@#SE,IDN@VAH@N2##SONE@0

Fig.5 Modified *shalaka* of Fig. 3. and encrypted messages.

Next, *shalaka* based encryption algorithm, as proposed above, for multiple messages is exemplified. Let us consider following multiple messages to be encrypted in the *shalaka* format

message1='Make AIR-FORCE ready to attack at 2 a.m tomorrow'
message2='The password of computer is floppy4.'
message3='Call me on 9344717 in urgency'
message4='We will capture first north side'
message5='If not OK tell infantry to attack '
message6='Reach spot23 with troops of 292005'

Among these messages, the number of characters in each message is different hence each message is padded with extra character '#' such that each message has number of characters equal to that of the longest message(here message1). Such an extra character can be randomly placed or inserted at the end of the message. Also, inter-word blank spaces are replaced by '@' for simplicity. The message matrix of size *6x49*, constructed from all the six messages, is shown in Fig. 6.

MESSAGE MATRIX

M A K E @ A I R - F O R C E @ R E A D Y @ T O @ A T T A C K @ @ A T @ 2 @ A . M @ T O M O R R O W
T H E @ P A S S W O # R D @ O F @ C O M # # # P U T E R # # @ I S @ F L # # O # P P # Y # # 4 .
C A L # # L @ M E # # # # @ # O N @ # # # 9 3 4 # # 4 7 1 7 @ # I N @ U # R G E N # # # C # # Y
W E @ W # # I # L L # @ C A P # # T U R E @ # F # # # I # R # S T @ # N # O R T # H # @ # S I D E
I # # F @ N # O T @ O K # @ T E L L @ # I # N # F A N T # R Y @ T O # @ A T T A C # # # # # K @ #
R E A # # # C # # H @ # # S P O T 2 3 # @ @ W I T # # H @ @ T R # O O P S @ # O F @ 2 9 2 0 # 0 5

Fig.6. Message Matrix obtained from six messages.

The total number of elements in the message matrix is 294 which is not a perfect square. Hence, square shaped *shalaka* is constructed using transposition, after adding extra letters, of elements in a matrix of size 18x18(the next nearest perfect square number) and is shown in the Fig.7. The rectangular *shalaka* of the dimension of message matrix is shown in Fig.8. The encrypted messages *row_mes, col_mes, diag_mes* taken along rows , columns and along diagonal respectively of square shaped *shalaka* are also shown below

row_mes ="M##WIRAT#E#EKHC@#AEEAWF#@@L#@#AP##N#IA#I#CRSL#O#-
S@LT#FWML@HOOE#O@R##@K#CR#C##ED#A@S@@#PTPRO@#EOEF##LTA@OTL2DCNU@3YO
@R##@M#EI@T##@#@O###NW@#9F#IAP3#FTTU4#A#TT##N#AE4ITHCR7##@K#1RR@@#7#YT@
@@S@RAI#TT#TSI@OO@@N##O2F@N@P@LU#ASA##OT@.#RRT#MOGTAO@#E#CFTPNH#@OP#
##2M##@#9OY###2R#CS#0R##IK#O4#D@0W.YE#5##############################"

col_mes =”MEIFCRDTAA@T@MORW##EAWROC#PE#SLOP#.##A#M#@N#347IUG#CY#WWIL
C#U@#I#@#T#SE#IF#@#E@#FTYOAA####R#CH#O3@THTOSO205#A@ROEEYOTC@@A@MR##T@
SODFO#UR@@#######LLE##@#47@N#E####E###A#R###S#O#@I###@OO@L#NA#@#TC#K##E##@
ST#W#@RO@F9###KAR@A@@TKA2.TOO##HPS#@@M#T#IF#PY4##C#@##O#9#1#@RN####@#L@
PTEF#RTNRH#D###NTKTLI#NRT@T##@##A###P2@I#@#P#@20##”

diag_mes = “#W#R.#O#Y#MPCE#@O#S##TLG##5#@SUT#0##A#I#A2R##AE7@AOM###T
P4#OS@####D#3IYOA##I##RC##TT@#E@K##CON@FH@@######FR@U#TC@NOC9O##IW##@@TR
@#TFO4##EAMCE3OU7S#@TY###ME#L#OY#4#@O.P#D###AI@#EO###R2#N#@##W#HEF@#A@AF
RH#0WFCOD#RN#KI@R#2I#RO###WT##NT@R@SEAL#@T1T@#A@L#@T@##RTPTL#OSAM9#R###
O@@@#FN@E@#R@OE####-##TIIEAS#PL@KP@@T2H#LKPC#T#@N###A”

Similarly encrypted message can also be obtained from rectangular s*halaka*.

Text-Shalaka(Square shaped)

M	#	#	W	I	R	A	T	#	E	#	E	K	H	C	@	#	A
E	E	A	W	F	#	@	@	L	#	@	#	A	P	#	#	N	#
I	A	#	I	#	C	R	S	L	#	O	#	-	S	@	L	T	#
F	W	M	L	@	H	O	O	E	#	O	@	R	#	#	@	K	#
C	R	#	C	#	#	E	D	#	A	@	S	@	@	#	P	T	P
R	O	@	#	E	O	E	F	#	#	L	T	A	@	O	T	L	2
D	C	N	U	@	3	Y	O	@	R	#	#	@	M	#	E	I	@
T	#	#	@	#	@	O	#	#	#	N	W	@	#	9	F	#	I
A	P	3	#	F	T	T	U	4	#	A	#	T	T	#	#	N	#
A	E	4	I	T	H	C	R	7	#	#	@	K	#	1	R	R	@
@	#	7	#	Y	T	@	@	@	S	@	R	A	I	#	T	T	#
T	S	I	@	O	O	@	@	N	#	#	O	2	F	@	N	@	P
@	L	U	#	A	S	A	#	#	O	T	@	.	#	R	R	T	#
M	O	G	T	A	O	@	#	E	#	C	F	T	P	N	H	#	@
O	P	#	#	#	2	M	#	#	@	#	9	O	Y	#	#	#	2
R	#	C	S	#	0	R	#	#	I	K	#	O	4	#	D	@	0
W	.	Y	E	#	5	#	#	#	#	#	#	#	#	#	#	#	#
#	#	#	#	#	#	#	#	#	#	#	#	#	#	#	#	#	#

Fig.7. Square shaped shalaka for the six messages

In order to decode *shalaka* one can start from any character element of the text *shalaka* and then concatenate all the 6th character in the forward direction. On reaching end element of the s*halaka* counting of the 6th character can be done either from the first element or towards left to the initially chosen character up to first character. 6th character taken from right are concatenated in right of the initial character and 6th character collected from left are concatenated to the left of initial character while keeping their order unchanged. For example, let us take *H*, encircled in Fig.7, as initial character. The collection of all the 6th letters, taken in succession, from *shalaka* from left and right side of *H* gives

REA###C##H@##SPOT23#@@WIT##H@TR#OOPS@#OF@2920#05#####

which is the last row of message matrix. Similarly, other rows can also be decrypted.

Text-Shalaka(rectangular)

M# # WI RAT # E# EK HC @# A EE AWF # @@L # @# AP # # N# I A# I # CRSL # O# -

S @L T # F WML @HOOE# O@R # # @K # CR # C # # E D# A @S @@# P T PRO@# E OEF

L T A @OT L 2 D CNU @3 Y O @R # # @M# EI @T # # @# @O # # # NW@# 9 F # I AP3

F T T U4 # A # TT # # N# AE 4 I T HCR 7 # # @K # 1 RR @@# 7 # YT @@@S @RA I # T

T # T S I @OO @@N # # O2 F @N @P @L U # A SA # # OT @. # RRT # MOGT A O @# E # C

F T P N H# @OP # # # 2 M# # @# 9 OY # # # 2 R# CS # 0 R # # I K# O4 # D @0 W. Y E # 5

Fig.8 Rectangular text shalaka for 6 messages.

The size of *shalaka* increases due to insertion of the extra characters which is necessary to make square representation. The size of the *shalaka* can be reduced by removing inter-word spacing. The message matrix and *shalaka* for the same set of six messages are shown below in Fig 9 and Fig.10 respectively.

MESSAGE MATRIX

MAKEAI R- FORCEREADYTOATTACKAT2A. MTOMORROW

THEPASSW# ORDOFCOM# # PUTE # RI SFL # O# PP # Y4 # .

CA # # # # LL # ME # # O# N# 9 3 4 # 4 7 1 7 I NURG# E # # N# C # # Y

WE WI # # L L # CAP # TUR # E # FI RST NORTH # # # # S # I # DE #

I F NOTO# KT EL # # # # LI N# FANT # R# YTOAT # TA # # CK

R # # # EACHSPOT 2 3 # WI T # HT # ROO # # PS # # OF 2 9 2 0 0 5 #

Fig.9 Message matrix with removed inter-word blank spaces.

Text-Shalaka(Square shaped)

M	#	C	W	#	R	A	T	A	E	I	#	K	H	#	W
F	#	E	E	#	I	N	#	A	P	#	#	O	E	I	A
#	#	T	A	#	S	#	L	O	C	R	S	L	L	#	H
-	W	L	#	K	S	F	#	#	C	T	P	O	O	M	A
E	O	R	R	E	P	L	T	C	D	#	#	#	2	E	O
#	T	#	3	R	F	O	U	#	#	E	C	#	R	#	W
A	O	N	#	#	I	D	M	#	E	L	#	Y	#	9	#
I	T	T	#	3	F	N	#	O	P	4	I	#	H	A	U
#	R	F	T	T	T	4	S	A	#	T	E	7	T	N	R
A	#	1	N	T	O	C	R	7	O	#	O	K	I	I	R
R	#	A	S	N	T	#	#	T	F	U	H	Y	#	2	L
R	#	T	P	#	#	G	#	O	S	A	O	#	#	A	#
.	#	#	#	T	#	M	P	E	#	#	O	T	#	#	S
T	F	O	P	#	#	A	2	M	#	N	I	#	9	O	#
#	#	#	2	R	Y	C	D	C	0	R	4	#	E	K	0
O	#	#	#	#	5	W	.	Y	#	#	#	#	#	#	#

Fig.10 Square shaped text *shalaka* of size *16x16* when inter-word blank spaces are removed.

The encrypted messages from this *shalaka* can also be obtained as above. The decryption requires tags to inter-word blank spaces for which first letters of the words, falling as the next neighbor of blank space, can be made capital/small and this information can be used in the decoding process to place inter-word spaces in a decrypted sentence. Similarly, rectangular *shalaka* can also be formed to generate encrypted messages. The message matrix and *shalaka* can also be rearranged along diagonals, permuted or manipulated in different ways to make very complex encryption of the message.

5. Conclusions

In this paper science of *SriRamShalaka*, one of the important tools used in the Vedic Astrology, is presented and taken as an ancient Indian method of text encryption based on matrix theory. Who first developed such representation of text in the ancient India is not known but it shows that there had been development of matrix theory in the ancient time. However, concerns of this paper have not been to obtain deep historical details of such developments. Here, *SriRamShalaka* has been taken as the method of text scrambling, based on modern theory of transposition, and its usefulness in text encryption has been investigated. Similar algorithms for the encryption/decryption of plain English text have been proposed and presented with examples. The proposed algorithms are useful in encrypting even smaller message. Further study can be made to develop algorithms for text encryption/decryption with combination of algorithms based on substitution. The relative strengths and weaknesses of encrypted messages obtained by different arrangements of message matrix are unexplored but very important for practical applications. The usability of proposed algorithm in text to image conversion and steganography can also be explored further.

Acknowledgments

Author is thankful to odd situations that pushed to study *SriRamShalaka*. The suggestions of unknown reviewers are also acknowledged which helped much in improving quality of the paper.

References

[1] A.Joseph Raphael, D. S. (2011). Cryptography and Steganography – A Survey. *nt. J. Comp. Tech. Appl., Vol 2 (3), ,* 626-630.
[2] Johnson N.F., e. (1998). Exploring steganography: Seeing the unseen. IEEE Computer, 1998, 31(2), 26-34. *IEEE Computer, 1998, 31(2), ,* 26-34.
[3] K., S. (2012). Cryptology and Communication Security (Editorial). *Defence Science Journal, Vol. 62, No. 1,*
[4] Kalyanaraman S. *Indus Script Cipher: Hieroglyphs of Indian Linguistic Area*, Sri Saraswati Research kendra, India
[5] Kak S(1988). The Aryabhata cipher, *Cryptologia*, vol. 12, 1988, pp. 113-117.
[6] Kak S.(2006). Aryabhatta's Mathematics, *RSA Conference* San Jose, Feb 13-17,2006.
[7] .M. Savari, a. M. (2012). All About encryption in smart card. *Proceeding of International conference on Cyber Security, Cyber Warfare and Digital Forensic (Cyber Sec).*
[8] Shannon, C. (1948). A mathematical theory of communication. *Bell Syst. Tech. J.,27* , 379-423.
[9] Shannon, C. A. (1948). Shannon, C.E. A mathematical theory of communication(part II). *Bell Syst. Tech. J.* , 623-56.
[10] Sharma, B. S. (January 2012). Steganographic Techniques of Data Hiding using Digital Images. *Defence Science Journal, Vol. 62, No. 1* , pp. 11-18.
[11] Stallings, W. (2005). *Cryptography and Network Security.* Prentice Hall, 2005.
[12] Tulsidas. (1563). *Sri Ram Charitmanas, 10th edn. Geeta Press, Gorakhpur.*