Volume 5, Issue 6, June 2015 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

# Key Generation Using Image Database for Inter Organizational Communication

**K. Vamsee Krishna**
Assoc. Professor in IT,
Sreenidhi Institute of Science and Technology
Hyderabad, India

**Pilla Dinesh**
Department of IT,
Sreenidi Institute of Science and Technology,
Hyderabad, India

***Abstract -- Our aim is to design and develop a full-fledged encryption technique for an organization that won't allow the attackers to decrypt the message in a stipulated amount of time. With the advent of technology and complex processing end systems, the attackers are decrypting the encrypted messages in a stipulated time. To avoid that situation, we use image[1] as a key to encrypt the message and also a time server to limit the data exposure for a certain amount of time. The properties of the image[2] such as pixel value, Hue, Saturation, Intensity etc., serves as keys to encrypt the message. The number of 8-bit keys depends on the number of characters present on the plain text as each key is used to encrypt each character present in the text or each pixel present on the image which makes the system complex and difficult to decrypt. The algorithm is purely a symmetric algorithm in which a database of uniform images is exchanged between the two parties; and the keys are selected based on the time.***

***This is mainly developed for inter-organizational communications as it includes database sharing and consumes more amounts of resources for generating the keys providing high security. As the data is the important financial asset for an organization it should be more secure. The algorithm provides the required security with the time constraint i.e. the receiver is provided with sufficient amount of time in which he can decrypt the encrypted text and if the time lapses the sender must resend the data.***

***Keywords -- Key Generation, Image data Extraction, Encryption, Decryption, Security Algorithm, Image Key Cryptography***

## I. INTRODUCTION

Over the Internet there are various communications such as the use of World Wide Web browsers or the e-Mail which may or may not be secure for sending and receiving information. The information sent by those means may include sensitive personal data which can be intercepted and will lead to heavy loss or may even threaten life. There is commercial activity going on the Internet and many organizations communicate with their clients that include sensitive information such as quotes, product information such as product formula, sales etc. To be able to do that, users would like to have a secure, private communication with the other party. Online users need a private and secure communications for other reasons as well. Hence the encryption algorithm survives on the time which is taken to break the cipher before the session expires.

The efficiency of the algorithm depends on the time taken to intercept the algorithm to extract the plain text from the encrypted text. My system takes a lot more time than the present existing algorithms, since the number of keys required to encrypt a single message are more than one and each character is encrypted with a different key which even increases the complexity in identifying the message from a large number of decrypts.

## II. EXISTING ALGORITHMS

All the existing algorithms make use of a finite set of keys and use direct strings or numbers as keys. The proposed system makes use of Image to generate keys. The properties of image such as RGB value, HSV value etc., serves as keys to encrypt the message which is very difficult to decrypt unless the attacker have the image dataset used for encryption.

## III. PROPOSED SYSTEM

The proposed system is as follows:

- Assume that the image database is exchanged with the trusted party in person or through a secure channel.
- The UNG or Unique Number Generator algorithm which is a part of "Time Extraction" is used to select an image along with the order of the service (properties of the image which depends on the pixel data of the image) from the image database will serve as a key to encrypt the data.
- The nine properties are extracted for each and every pixel of the selected image and the order of the service is selected using the unique number and will serve as keyset. The first nine keys are extracted using the nine properties of the image and so on.
- Now each key is used to encrypt each character or each pixel value in the image.

The System architecture is as shown in the Figure 1.

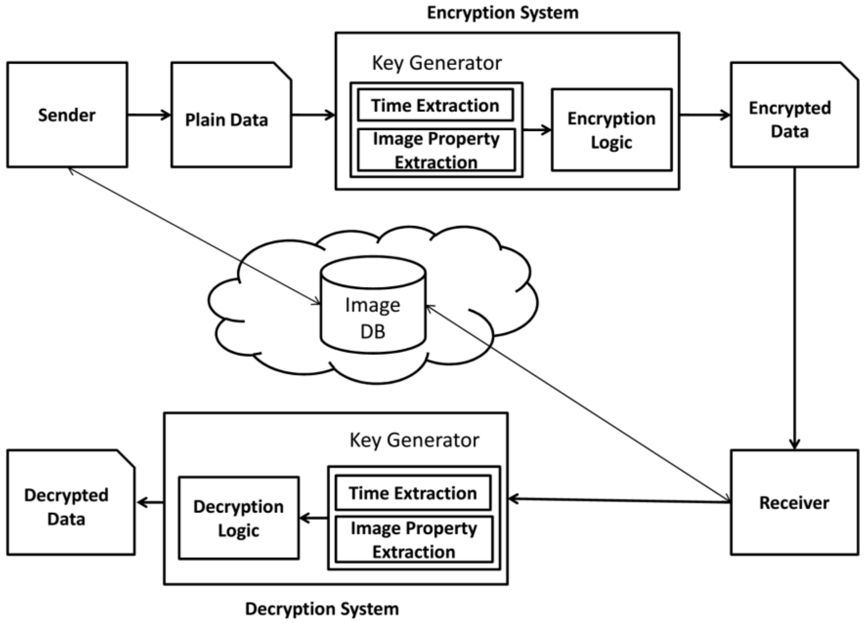


Figure 1: System Architecture

***A. UNG Algorithm:***

The UNG algorithm is as follows:

1. The UNG algorithm collects the time from one of the following servers:
   i. NIST, Boulder, Colorado ATOMICTIME_SERVER="132.163.4.101";
   ii. NIST, Gaithersburg, Maryland ATOMICTIME_SERVER="129.6.15.28";
   iii. NIST, Gaithersburg, Maryland ATOMICTIME_SERVER="129.6.15.30";
2. The unique number is generated using the following time format which is obtained from the server
   Server Time: 57159 15-05-17 17:40:11 50 0 0 193.4 UTC (NIST)
3. From the Server time the numbers were collected in the same order leaving the others from half part of the minutes as follows 57159150517174, which helps in providing the relaxation time for decrypting the text.
4. Now manipulate the data in such a way that it will produce a 7-digit decimal number.
   The example of the unique number that is generated from the UNG algorithm
   The Unique number generated with the Time server with the IP 129.6.15.30 is as follows:
   6233089
   The image property services used are as follows:
   i. HMMD Service
   ii. HSL Service
   iii. HSV Service
   iv. LAB Service
   v. LUV Service
   vi. RGB Service
   vii. XYZ Service
   viii. YCbCr Service
   ix. YUV Service

**Note:** For encrypting we just used few permutations, substitutions along with an XOR as a basic encryption algorithm as our aim is to provide an approach to secure the information.

***B. Relaxation Time for Decryption:***

The max and min time to decrypt the encrypted text is
Max time: 20min – execution time to encrypt.
Min time: 10min
This is achieved using the Time Extraction and Unique Number Generator.

***C. Complexity Analysis:***

The complexity of the algorithm lies in two areas first one is the complexity in calculating the keys from the image and the other is the encryption algorithm used in this algorithm the complexity is as follows:
O(P)+O(C)
Where P is the number of pixels in the image and C is the number of characters present on the image.

***D. Space analysis based on the algorithms used for the encryption:***

The spaces occupied by the algorithms such as DES[3], Triple DES[3] and XOR[4] are as shown in the Table 1.

Table1: Space comparison of different encryption algorithms using our approach.

| Encryption Technique | Plain Text | Encrypted Text | Decrypted Text |
|---|---|---|---|
| **XOR** | 160 KB | 160 KB | 160 KB |
| **DES** | 160 KB | 218 KB | 160KB |
| **T-DES** | 160KB | 390 KB | 160KB |

The results of encrypting the same file using different encryption techniques and their space analysis[5] is as mentioned in the table 1 and their comparisons is diagrammatically calculated as shown in the Figure 2.

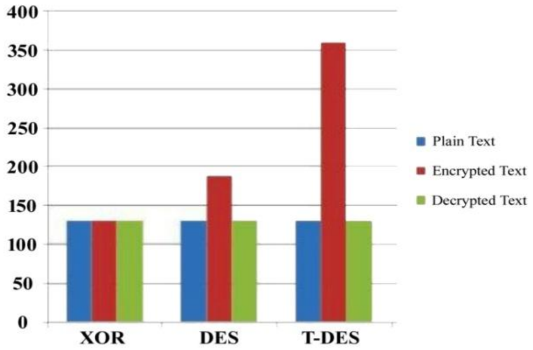


Figure 2: Space comparison of different encryption algorithms using our approach.

***E. Time analysis based on the algorithms used for the encryption:***

The Time taken [6] by the algorithms such as XOR (with our approach), DES and triple-DES are as shown in the Table 2.

Table 2: Time comparison of different encryption algorithms using our approach.

| Input Size (KB) | XOR with our approach | DES | T-DES |
|---|---|---|---|
| **19 KB** | 58 sec | 24 sec | 59 sec |
| **35 KB** | 111 sec | 68 sec | 208 sec |
| **40 KB** | 123 sec | 77 sec | 223 sec |

The results of encrypting the same file using different encryption techniques and their time analysis is as mentioned in the table 2 and their comparisons is diagrammatically calculated as shown in the Figure 3.

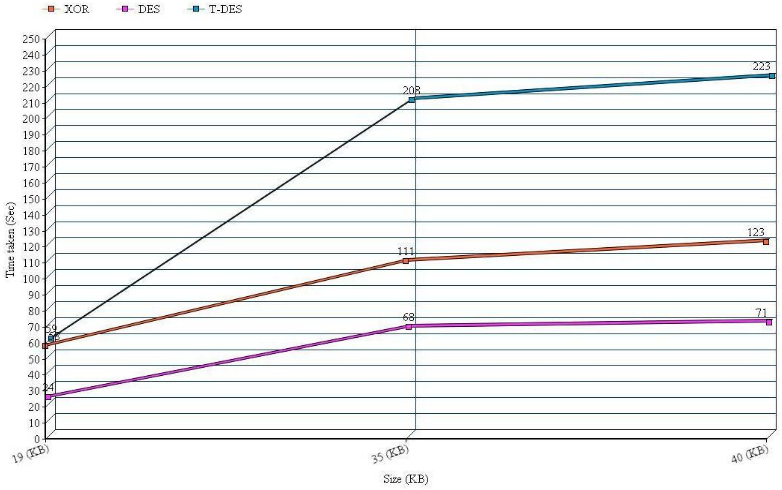


Figure 3: Time comparison of different encryption algorithms using our approach.

## III. EXPERIMENTAL SETUP AND RESULTS

The experimental environment and results are as follows:

***A. Experimental setup:***

We were working in the following environment:

1. Windows 8 operating system.
2. JDK 1.8.
3. Intel atom 1.6 GHz Processor.

4. 2GB RAM.
5. IDE used: Eclipse JEE KEPLER
6. Other requirements: Uniform Image database (for experimenting we took 99 images)

***B. Experimental Results:***

The experimental results are as follows:

1. Browsing for the image database.

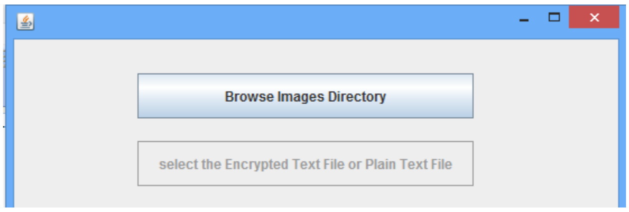
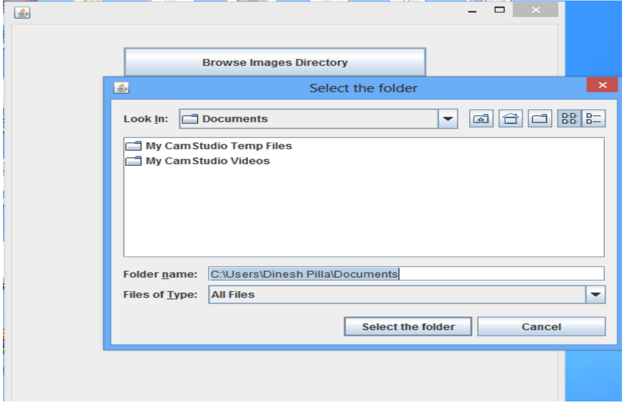


Figure 4: Browsing image database

2. Once the image database is selected for the experimental purpose the images are shown as follows.



Look In: Documents

Folder name: C:\Users\Dinesh Pilla\Documents

Files of Type: All Files

Figure 5

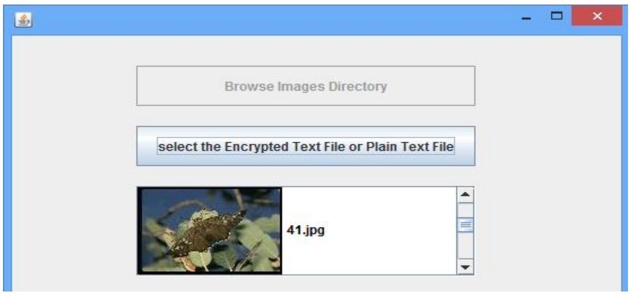
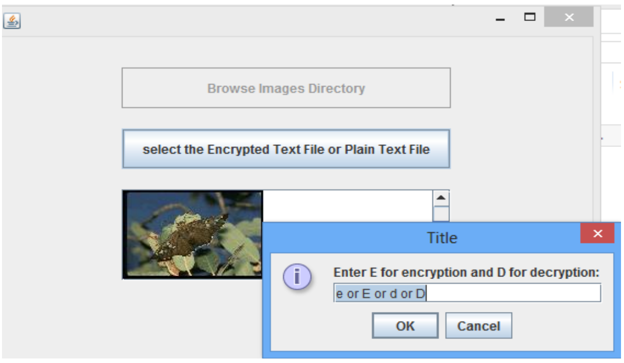


Figure 6

3. Now select the plain text to encrypt the text press e/E or if you have the encrypted text then select encrypted text in-order to decrypt press d/D.

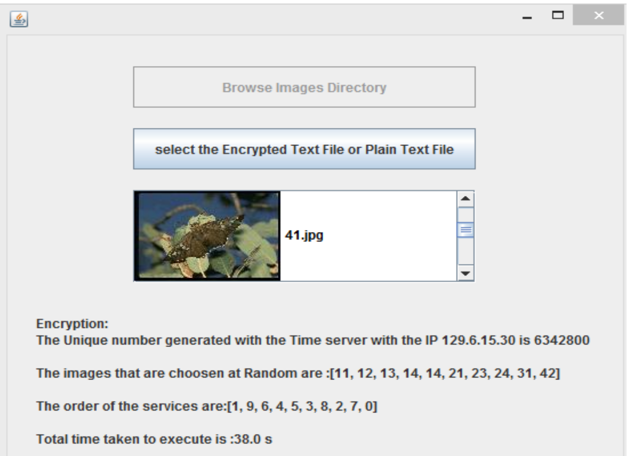


Enter E for encryption and D for decryption:

e or E or d or D

Figure 7

4. Now the encrypted text is placed in the same directory of the plain text with the name encrypt if you encrypt the text, if you decrypt the text then the decrypted text is placed in same directory of the encrypted text with two files with the names decrypt0 and decrypt1. The plain text will be in decrypt0 if the encrypted text is decrypted in 10 min – encrypted time – digital second min time else will be in decrypt1.



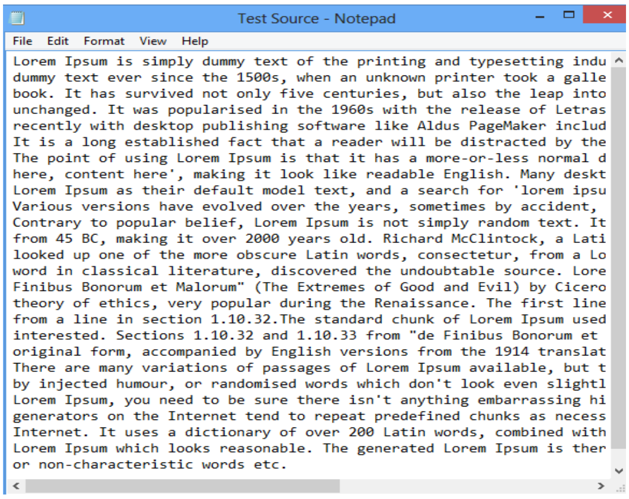
Encryption:
The Unique number generated with the Time server with the IP 129.6.15.30 is 6342800

The images that are choosen at Random are :[11, 12, 13, 14, 14, 21, 23, 24, 31, 42]

The order of the services are:[1, 9, 6, 4, 5, 3, 8, 2, 7, 0]

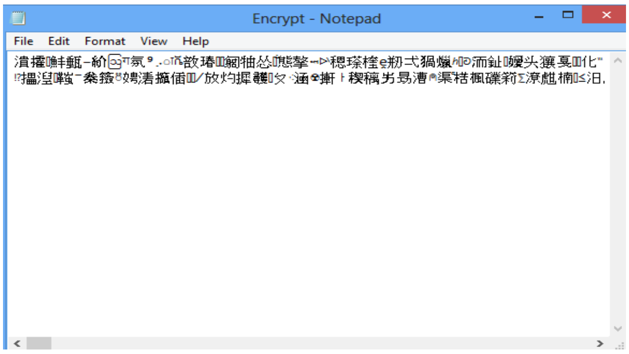
Total time taken to execute is :38.0 s

Figure 8



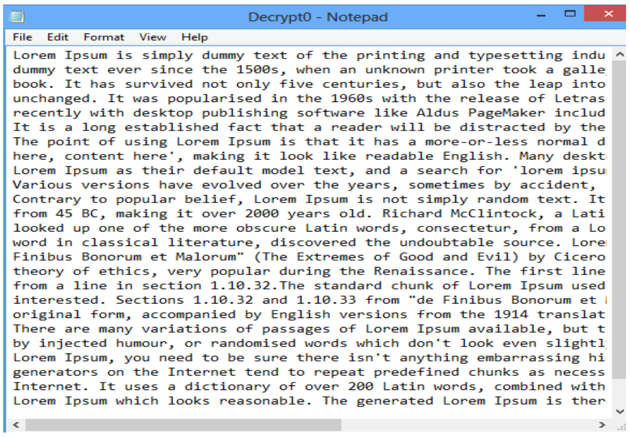
```
Lorem Ipsum is simply dummy text of the printing and typesetting indu
dummy text ever since the 1500s, when an unknown printer took a galle
book. It has survived not only five centuries, but also the leap into
unchanged. It was popularised in the 1960s with the release of Letras
recently with desktop publishing software like Aldus PageMaker includ
It is a long established fact that a reader will be distracted by the
The point of using Lorem Ipsum is that it has a more-or-less normal d
here, content here', making it look like readable English. Many deskt
Lorem Ipsum as their default model text, and a search for 'lorem ipsu
Various versions have evolved over the years, sometimes by accident,
Contrary to popular belief, Lorem Ipsum is not simply random text. It
from 45 BC, making it over 2000 years old. Richard McClintock, a Lati
looked up one of the more obscure Latin words, consectetur, from a Lo
word in classical literature, discovered the undoubtable source. Lore
Finibus Bonorum et Malorum" (The Extremes of Good and Evil) by Cicero
theory of ethics, very popular during the Renaissance. The first line
from a line in section 1.10.32.The standard chunk of Lorem Ipsum used
interested. Sections 1.10.32 and 1.10.33 from "de Finibus Bonorum et
original form, accompanied by English versions from the 1914 translat
There are many variations of passages of Lorem Ipsum available, but t
by injected humour, or randomised words which don't look even slightl
Lorem Ipsum, you need to be sure there isn't anything embarrassing hi
generators on the Internet tend to repeat predefined chunks as necess
Internet. It uses a dictionary of over 200 Latin words, combined with
Lorem Ipsum which looks reasonable. The generated Lorem Ipsum is ther
or non-characteristic words etc.
```

Figure 9 - Plain Text



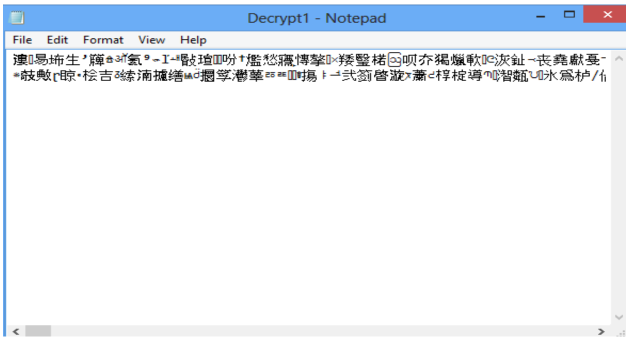
```
湞擢□鲑甀–紒ⓢ꜒氛⁹ .·ㅇ卬敨瑃□䥧牰怂□態挐↔▷穏瑽楏ɐ刟弌猧爈ℎ□ⓞ洏鉎□嫂头瀷戛□化"
⁉擂湟□䗐¯粊鐩ᵒ娉渚攟偛□/放灼揘䘿□夊·涵☢揦ㅏ稧稨㚐昜漕ⁿ渠桔楓磼箣Σ潦䶈楠□≤汨.
```

Figure 10 - Encrypted text



```
Lorem Ipsum is simply dummy text of the printing and typesetting indu
dummy text ever since the 1500s, when an unknown printer took a galle
book. It has survived not only five centuries, but also the leap into
unchanged. It was popularised in the 1960s with the release of Letras
recently with desktop publishing software like Aldus PageMaker includ
It is a long established fact that a reader will be distracted by the
The point of using Lorem Ipsum is that it has a more-or-less normal d
here, content here', making it look like readable English. Many deskt
Lorem Ipsum as their default model text, and a search for 'lorem ipsu
Various versions have evolved over the years, sometimes by accident,
Contrary to popular belief, Lorem Ipsum is not simply random text. It
from 45 BC, making it over 2000 years old. Richard McClintock, a Lati
looked up one of the more obscure Latin words, consectetur, from a Lo
word in classical literature, discovered the undoubtable source. Lore
Finibus Bonorum et Malorum" (The Extremes of Good and Evil) by Cicero
theory of ethics, very popular during the Renaissance. The first line
from a line in section 1.10.32.The standard chunk of Lorem Ipsum used
interested. Sections 1.10.32 and 1.10.33 from "de Finibus Bonorum et
original form, accompanied by English versions from the 1914 translat
There are many variations of passages of Lorem Ipsum available, but t
by injected humour, or randomised words which don't look even slightl
Lorem Ipsum, you need to be sure there isn't anything embarrassing hi
generators on the Internet tend to repeat predefined chunks as necess
Internet. It uses a dictionary of over 200 Latin words, combined with
Lorem Ipsum which looks reasonable. The generated Lorem Ipsum is ther
```

Figure 11 - Decrypt0



```
漣□易坲生’饢ㅿ㆒氯⁹⌐Ⅰ⁻骰瑄□吩†尷愸癘愽摯□×㨪鑒楉□呗夰猲爈軟□泧釒¬丧堯猷戞⁻
⁎菝敷ɼ晾•桧吉綊湳攎缮擱箰灪蘳□暘⊦⁻弐劄朁漩ˣ蕭桴椗導□㴬甀□氺爲栌/佮
```

Figure 12 - Decrypt1

## IV. CONCLUSION AND FUTURE SCOPE

Since each character is encrypted with a separate key it is very difficult to decrypt the message. Hence it is very reliable and secure form of encryption. With the advent of technology the encryption techniques must move from a finite set of keys to encrypt the text to large number of keys which make the system complex but secure.

## REFERENCES

[1] Image Encryption using color key images, S Reddy Jyoteeswara Prasad and R V S Sathyanarayana, S V University College of Engineering, Tirupati, Andhra Pradesh, India, available at http://www.ijeetc.com/

[2] lyer K C and Subramanya A (2009), “Image Encryption by Pixel Property Separation”, Cryptology.

[3] Data Encryption Standard (1999), National Institute of Standards and Technology, available at http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf

[4] Han J W, Park C-S, Ryu D-H and Kim ES (1999), “Optical Image Encryption Based on XOR Operations”, Optical Engineering, Vol. 38, No. 1, pp. 47-54.

[5] A Comparative Analysis of Encryption Algorithms for Better Utilization, by Anuj Kumar, Sapna Sinha and Rahul Chaudhary, Amity University Noida available at http://www.ijcaonline.org/archives/volume71/number14/12426-8934

[6] A fast encrypting algorithm by Ritu Agarwal, Dhiraj Dafouti, Vishal Bhargava, Nikhil Maheshwari, Shobha Tyagi Department of Information Technology, available at http://www.researchgate.net/publication/267839980_A_FAST_ENCRYPTING_ALGORITHM