Comparative Analysis of NPN Algorithm & DES Algorithm

Mrs. Mukta Sharma#, Dr. R B Garg, Professor*, Ms. Surbhi Dwivedi#

#Research Scholar, TMU, *Tecnia Institute of Advanced Studies, #Sr. Software Engineer, NIIT Technologies Ltd.

[1]m.mukta19@gmail.com

#NH 24,bagadpur,Delhi Road, Moradabad, Uttar Pradesh, India

[2]garg1943@gmail.com

*Madhuban Chowk, Sector 14, Rohini, New Delhi, India

[3]leosurabhi@gmail.com

#Plot No. TZ 2 & 2A, Sector Tech Zone, Greater Noida, Uttar Pradesh, India

***Abstract*- Various sectors such as such as the retail, hospitality, banking and financial introduced information technology years back. Ongoing enhancements have always offered a wider scope of growth to these sectors. Zooming in the Banking & Financial sector most banks are now offering the Online banking services. The objectives of introducing the e-banking services were profits, fast service, improved productivity, customer satisfaction, 24x7 operations & cost savings. As much the growth of internet probes the customers to use the new online services & banks to offer the same; equally it makes the customers & business skeptical about the security being implemented. The business needs to ensure the security of each electronic transaction over the internet. Critical areas to focus for this are secure communication channel, third party for database maintenance & a robust non-breachable data encryption technique. Any online transaction constitutes of confidential data. Any damage to that data or hacking of that data may bring significant amount of loss to both the parties involved.**

Cryptography is a widely used science for secret writing. Encryption is one process of secret writing. The process involves a key and an algorithm to generate a Cipher text (secret code) from a plain text. On the other hand, decryption helps to retrieve the original text using the same key. Now what is the key here? The Key is the core string (word) being used in the algorithm. It is kept private. Cryptography has classified the keys being used in the process as Symmetric & Asymmetric. Symmetric key algorithms are most commonly used type where a single key is used for both encryption and decryption [4]. In asymmetric key algorithms different keys are used both for encryption and decryption.

This paper highlights on designing & implementing a symmetric key encryption algorithm for securing the online transactions using the concept of Multithreading to optimize the time. The work has been implemented using JDK 1.6 as the programming environment. A comparative analysis of DES and the proposed algorithm has been depicted with the help of table and various graphs.

***Keywords*— Cryptography, Symmetric & Asymmetric Cryptography, Plain Text, Cipher Text, Encryption, Decryption, Multithreading. NPN, DES**

I. INTRODUCTION

The banking sector has seen major advancements by the ongoing information technology innovations. Bankers have explored new means to deliver their services to the customers. E-banking offers the prospect for easy access to the banking activities like retrieve an account balance, fund transfers etc. With Internet, the world witnesses the escalation in banking industries to facilitate electronic payments. However, the businesses are more subject to threats of malicious activities and cyber-crimes. Cyber criminals are the people involved in destructive activities. They use computers & other technology for the same. They can be inside the system or a may be far distant outsider. The only thing common amongst them is the thought process, the willingness to breach the security, to have unauthorized access, to acquire others confidential data & many more. To successfully design & implement security we need to be a step ahead or perhaps think on the same line as the cyber criminals do.

A. *Cryptography*

Every successful theft questions the security measures taken & implemented in the particular system. Various researches are going on in the field of cryptography to ensure security of the confidential data across the internet. Cryptography as can be seen in the history of mankind is not a new science. The very elementary use of this science was done long back in the time of Julius Ceaser (100-44 BC).

Being derived from two Greek words Crypto (Secret) & graphs(writing), the term cryptography explains itself as a science encompassing the principles & methods of transforming plain text into a secret coded text. Cryptography is used to protect the data from theft or alteration, user authentication etc. There are 3 types of cryptography techniques: Secret key, Public Key & Hash function. In all cases, the initial unencrypted data is referred to as plaintext. It is encrypted into cipher text, which will in turn (usually) be decrypted into usable plaintext. [2]

978-1-4799-6896-1/14/$31.00 ©2014 IEEE

1. *Purpose of Cryptography: Cryptography can be used to provide:*
   - Confidentiality – only authorised recipient can read data.
   - Data integrity - ensures no data alteration occurred.
   - Authentication - ensure data originated from a particular party
   - Non-repudiation: A mechanism to prove that the sender really sent this message. [2]
2. *Types of Cryptography*
   - **Secret key or Symmetric key:** In this sender and receiver possess the same single key. It can be divided into stream ciphers and block ciphers. Stream cipher encrypts a single bit of plain text at a time, whereas block cipher encrypts a number of bits as a single unit.
   - **Public key or Asymmetric key:** Involves two related keys called a key-pair: one public key known to anyone and one private key that only the owner knows.
   - **Hash Functions:** Uses a mathematical transformation to irreversibly "encrypt" information

B. *Scope of the paper*

This paper focuses on designing and implementing a symmetric key encryption algorithm that is NPN (nth Prime number) for securing the online transactions. The algorithm has been implemented in Eclipse IDE, Java 1.6 on Dell Laptop with Configuration: Intel Core i5 @2.60GHz 4GB RAM and 64 bit Windows 7 OS using the core concept of Multithreading to optimize the time. A comparative analysis of DES and the proposed algorithm has been depicted with the help of table and various graphs. DES is the first encryption standard by NIST (National Institute of Standards and Technology). While designing NPN, DES was set as a minimum benchmark to be achieved.

C. *Outline of the paper*

The paper is strategically organized in 5 sections. First section gives introduction about cryptography and its various techniques. Second comprises of the proposed encryption and decryption algorithm. Second section covered the benefit of using the concept of Multithreading. Later section explains the implementation of DES algorithm in Java, followed by a comparative analysis between the proposed algorithm and DES algorithm. Finally, summarizing with a conclusion.

II. NPN ALGORITHM

A. *Encryption - Pseudo code*

```
SET nthprime=0
SET Key=RandomNumber
GET Key = DETERMINE(AbsoluteValue(Key))
INIT Ciphertext =0
READ plaintext
FOR 1 to sizeof(plaintext)
    SET nthVal = ascii(plaintext[i])
    SET P=Key, count = 0
    FOR P to count!=nthVal
            SET status=1
            IF P==2
            nthprime=i
            count++
            ELSE IF P%2==0
            nthprime=0
            ELSE
            FOR  j = 3 j <= Math.sqrt(i)  j+=2
            IF  P%j == 0

            status = 0;
            BREAK
            IF status != 0
              nthprime=i;
              count++;
              status = 1;
            FOREND
    retStr.ADD(nthprime);
    FOREND
    Ciphertext = nthPrime+ Constant
```

Add a Constant to the Obtained value to give a Cipher Text. Constant can be the last 2 digits of the Key. As we go higher in numbers the Prime Numbers turn sparse. The notion to add a Constant to the obtained prime number is to offer a larger set of Natural Numbers.

B. *Decryption - Pseudo code*

```
READ cipher
INIT i=0;
INIT count;
INIT nthprime=0;
FOR SET i=key, count=0 nthprime!=cipher
  INIT status=1
  IF(i==2)nthprime=i
  count++
  ELSEIF i%2==0
            nthprime=0
            ELSE
            FOR SET j = 3 , j <= Math.sqrt(i), j+=2
            IF i%j == 0
            status = 0
            BREAK
            FOREND
            IF status != 0
                    nthprime=i
                    count++
            status = 1
return plaintext.
```

III. FAST NPN ALGORITHM

The current algorithm effectively and efficiently uses the concept of Multithreading, which leads to optimize the time for encrypting as well as decrypting the data. Multithreading is a type of multitasking. A multithreaded code has more than one section that can run parallel to each other. Java allows us to write efficient programs that makes best use of the CPU. Multithreading in java can be implemented using either the Thread Class or the Runnable Interface.

The String, StringBuffer, and StringBuilder classes are used in the proposed algorithm. These classes are defined in java.lang. String object is immutable (String object cannot be changed, which means every time you are making any change a new string object is being created) whereas StringBuffer objects are mutable (can make changes to the value stored in the object). StringBuilder is also same like StringBuffer (mutable) except StringBuffer is not synchronized as in one cannot use it with Multithreading.

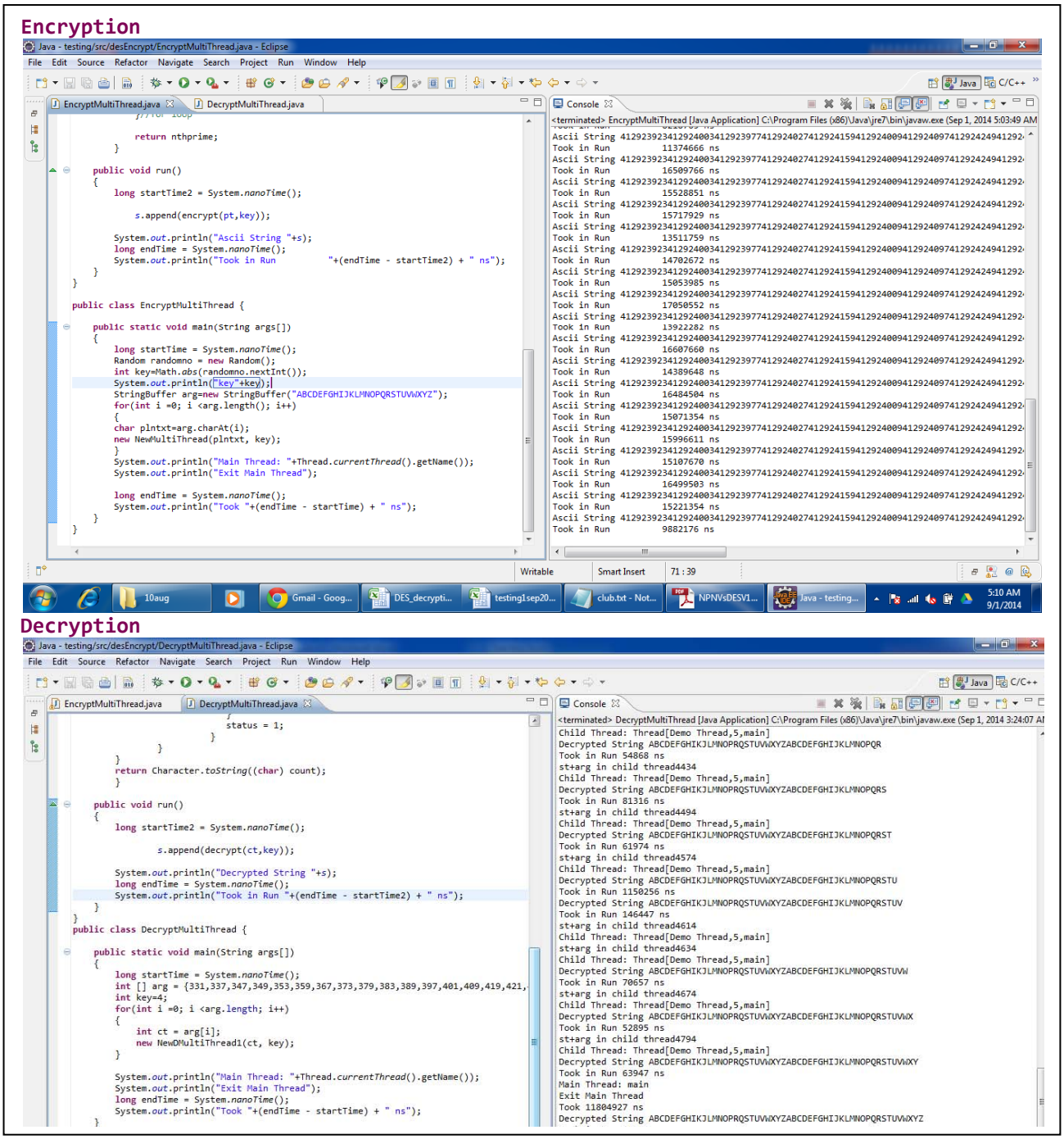

FIG. 1 SCREEN SHOTS OF JAVA CODE IN ECLIPSE IDE.

IV. DES ALGORITHM

DES is a symmetric block cipher encryption algorithm developed by IBM. It was standardized in 1977. The algorithm encrypts 64 bits size blocks with a key of size 56 bits. It performs 16 rounds which all perform the similar operation. However, a distinct sub-key is obtained in each round from the core key.

```
import java.security.InvalidKeyException;
import java.security.NoSuchAlgorithmException;
import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;
import javax.crypto.KeyGenerator;
import javax.crypto.NoSuchPaddingException;
import javax.crypto.SecretKey;

public class DESEncryptionJava
{
        public static void main(String[] argv) {

                long startTime = System.nanoTime();
                try{

                KeyGenerator keygen =
KeyGenerator.getInstance("DES");
                SecretKey pvtDesKey =
keygen.generateKey();

                Cipher desCipherText;
  // Create the cipher
                desCipherText =
Cipher.getInstance("DES/ECB/PKCS5Padding");

                // Initialize the cipher for encryption

desCipherText.init(Cipher.ENCRYPT_MODE,
pvtDesKey);

                //sensitive information
 byte[] text =
"ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHI
JKLMNOPQRSTUVWXYZABCDEFGHIJKLMNOPQR
STUVWXYZABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJ
KLMNOPQRSTUVWXYZABCDEFGHIJKLMNOPQRS
TUVWXYZABCDEFGHIJKLMNOPQRSTUVWXYZA
BCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJK
LMNOPQRSTUVWXYZ".getBytes();
                System.out.println("Text in Byte
Format : " + text);
                System.out.println("Text : " + new
String(text)); // Encrypt the text
byte[] encryptedText = desCipherText.doFinal(text);
                System.out.println("Text : " + new
String(encryptedText));
                System.out.println("Text Encryted : " +
encryptedText);
                // Initialize the same cipher for
decryption

desCipherText.init(Cipher.DECRYPT_MODE, pvtDesKey);
          // Decrypt the text
                byte[] decryptedText =
desCipherText.doFinal(encryptedText);
   System.out.println("Text Decryted : " + new
String(decryptedText));
}catch(NoSuchAlgorithmException e){
                e.printStackTrace();
        }catch(NoSuchPaddingException e){
                e.printStackTrace();
        }catch(InvalidKeyException e){
                e.printStackTrace();
        }catch(IllegalBlockSizeException e){
                e.printStackTrace();
}catch(BadPaddingException e){
                e.printStackTrace();
        }
        long endTime = System.nanoTime();
        System.out.println("Took "+(endTime -
startTime) + " ns");          }}
```

[3]

V. COMPARISON

Comparing both the algorithms, DES at core has 14, 2-dimensional arrays of sizes varying from 4×8 to 8×8 & 16×4 contributing memory size of 824 bits. However, the Java implementation shows it to use 9 class objects, thus 24 bytes (size of object) × 9. That is 216 bytes & the size of plain text. The NPN algorithm on the other hand, uses only 3 classes & 6 integer data types, contributing to total of 148 bytes & 2×size of text. Further here, we can see a table is created quoting the number of characters being entered as a String to be encrypted. The table also highlights the time taken by both DES & NPN algorithms respectively. On the basis of the table values we have created the graphs also.

TABLE I
NPN VS DES – ENCRYPTION TIME

Character Input	DES (Nano Seconds)	NPN - Multithread (Nano Seconds)
2	282748646	15248985
26	139799531	29959946
52	142914775	29573502
78	141157812	137772964
156	147400141	763918039

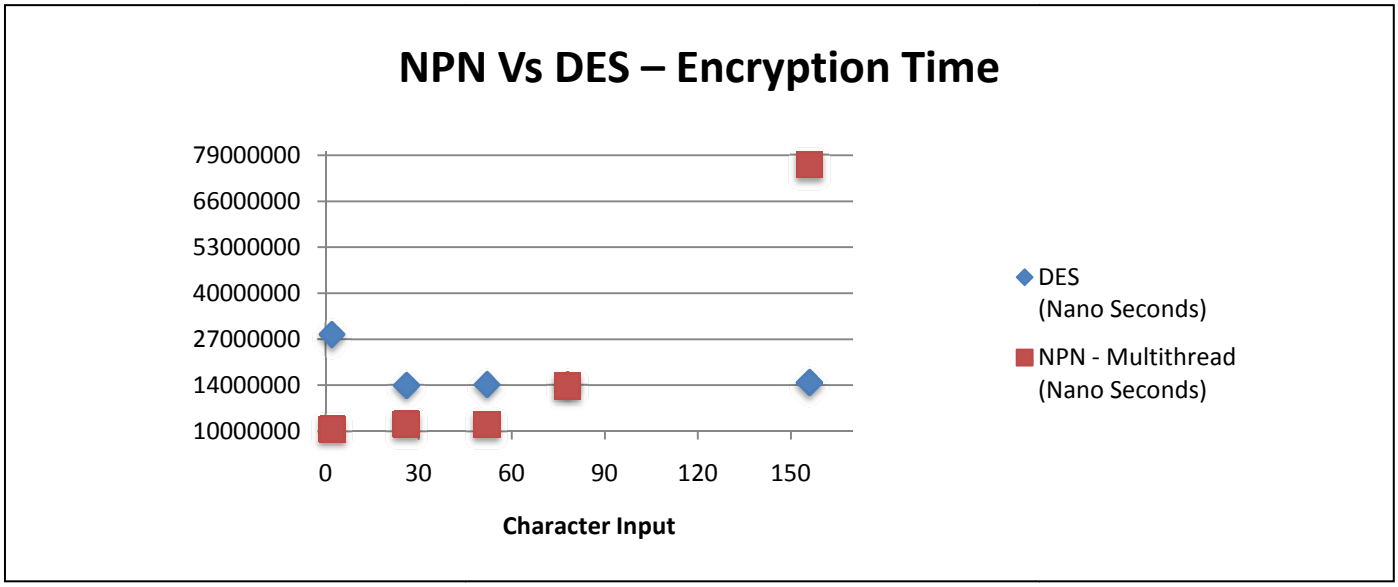

FIG. 2 GRAPHICAL DISPLAY OF ENCRYPTION TIME BEING TAKEN

TABLE II
NPN VS DES – DECRYPTION TIME

Character Input	DES (Nano Seconds)	NPN - Multithread (Nano Seconds)
2	193161700	1141967
26	218222992	5988515
52	194184457	11836506

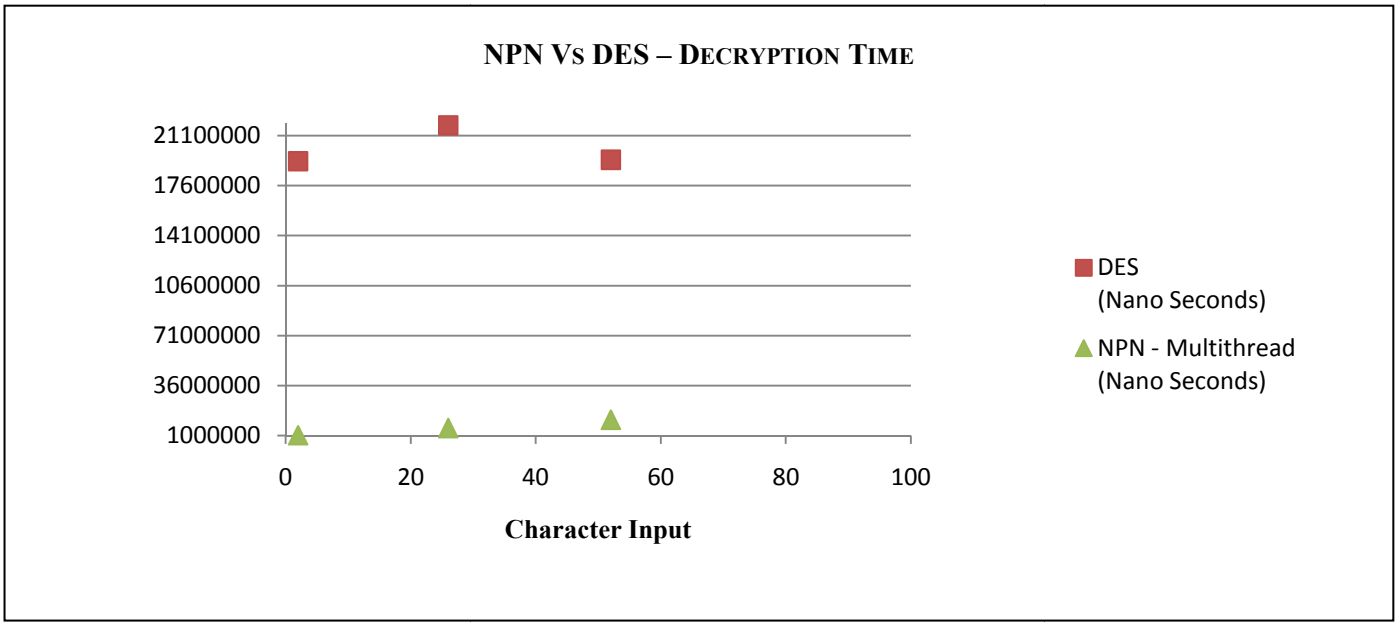

FIG. 3 GRAPHICAL DISPLAY OF DECRYPTION TIME BEING TAKEN

VI. CONCLUSIONS

Each algorithm has its own advantages and disadvantages, our system proposed a good strategy of making most out of the advantages of prime numbers and ascii values. The developed system could be used in any network services for network security. The Space complexity has also been dealt as an essential objective to be met in this algorithm.

In cryptography, key size or key length is the size measured in bits of the key used in a cryptographic algorithm (such as a cipher). An algorithm's key length is distinct from its cryptographic security, which is a logarithmic measure of the fastest known computational attack on the algorithm, also measured in bits. The security of an algorithm cannot exceed its key length (since any algorithm can be cracked by brute force), but it can be smaller. Most symmetric-key algorithms in common use are designed to have security equal to their key length.[1] The proposed algorithm is based on 32 bit key generation for a 16 bit plain text. Hence meeting the minimum requirement of the symmetric-key algorithm key generation factor.

If we see the table and graph that depicts Encryption time. We can observe that the NPN algorithm takes relatively less time as compared to DES till the length of the string is less. However, as the length is raised more than 100 the algorithm faces the time complexity. Hence, the problem left to be resolved is the time complexity of this algorithm. As we can very well see the table & the graphs depicting time taken by NPN for Encryption & Decryption is substantially high as compared to DES. The algorithm being based on Prime Number itself makes it more on Time complexity.

VII. FUTURE SCOPE

The work related to proposed algorithm offers a future scope on work around decreasing the encrypting and decrypting time. The concept of encryption using multithreading technique enhances the speed of encryption system but still the generation of Prime Number is a time consuming task.

REFERENCES

[1] A.S. Tanenbaum, “The Application Layer”, Computer Networks, Ed. New Delhi: PHI, 1996, pp. 577-766

[2] G. C. Kessler, “An Overview of Cryptography”, 1999 Edition of Handbook on Local Area Networks, published by Auerbach in September 1998.

[3] M Yong. (2008). JCE Encryption – Data Encryption Standard (DES). [Web log comment]. Retrieved from http://www.mkyong.com/java/jce-encryption-data-encryption-standard-des-tutorial/]

[4] Naveed, I. and Puech, W. (2012) Data Cryptography, in Signal and Image Processing for Biometrics (eds A. Naït-Ali and R. Fournier), John Wiley & Sons, Inc., Hoboken, NJ, USA. doi: 10.1002/9781118561911.ch13