A NUMERICAL METHOD BASED ENCRYPTION ALGORITHM WITH STEGANOGRAPHY

Amartya Ghosh and Anirban Saha

Regent Education & Research Foundation, Barrackpore, West Bengal, India.
com.amartya@gmail.com and maths.anirban@gmail.com

ABSTRACT

Now-a-days many encryption algorithms have been proposed for network security. In this paper, a new cryptographic algorithm for network security is proposed to assist the effectiveness of network security. Here symmetric key concept instead of public key is considered to develop the encryption – decryption algorithm. Also, to give more security in the algorithm, the idea of one way function alongwith Newton's method is applied as a secret key to the proposed work as well as Digital Signature Standard (DSS) technology is used to send the key. Moreover, steganography is used to hide the cipher within a picture in encryption algorithm. In brief, a numerical method based secret key encryption – decryption algorithm is developed using steganography to enhance the network security.

KEYWORDS

Encryption, Decryption, Cryptography, Numerical Method, Steganography.

1. INTRODUCTION

At the present time sending messages in hidden form plays a vital role in Network Security. Cryptography is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication. Actually, it is the study of methods of sending messages in hidden form so that only intended recipients can remove the disguise and read the message. Cryptography [1, 2] is not the only means of providing information security, but rather one set of techniques. Cryptography offers efficient solution to protect sensitive information in a large number of applications including personal data security, internet security, diplomatic and military communications security, etc. through the processes of encryption/decryption. It is traditionally the study of means of converting information from its normal, comprehensible form into an incomprehensible format, rendering it unreadable without secret knowledge-the art of encryption. Another technique to hide or secure data is Steganography. There are several ways to hide data using this technique like steganography in image, steganography in video, steganography in audio, steganography in document etc.

A cryptosystem is a set of algorithm, indexed by key(s), for encoding messages into cipher text and decoding them back into plaintext [3]. The model for a secret key system is first proposed by Shannon [4]. The original information, which is to be protected by cryptography, is called plaintext. Encryption is the process of converting plaintext into an unreadable form termed cipher text, or occasionally, a cryptogram. Decryption is the reverse process, recovering the plaintext back from a cipher text.

Rupak Bhattacharyya et al. (Eds) : ACER 2013,
pp. 149–157, 2013. © CS & IT-CSCP 2013 DOI : 10.5121/csit.2013.3214

Cryptographic algorithms are categorized based on the number of keys that are employed for encryption and decryption, and further defined by their application and use. The two types of algorithms are Secret Key Cryptography or Symmetric Key Cryptography (SKC) and Public Key Cryptography or Asymmetric Key Cryptography (PKC). In case of Symmetric Key Cryptography, encrypt key and decrypt key are symmetric and the examples of this kind of algorithm are DES, AES etc. Also, in case of Asymmetric Key Cryptography, encrypt key and decrypt key are not same and the examples of this kind of algorithm are RSA, McEliece etc.

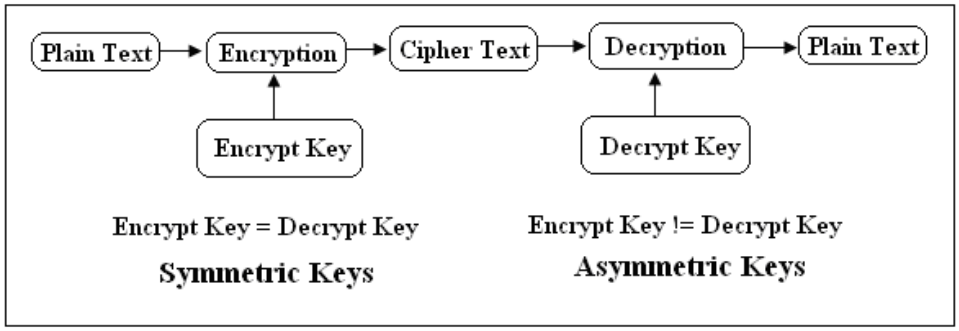

Figure 1: Basic Idea of Cryptography

Till now, many encryption – decryption algorithms [5, 6] based on symmetric key and asymmetric key have been proposed. Algorithms on Symmetric Key Cryptography are the most commonly used cryptography type before invention of public key encryption. One application of the symmetric key or the private key cryptography is DES (Data Encryption Standard) which is widely used in the system. Also, due to a number of drawbacks in the encryption – decryption algorithms on asymmetric key like RSA etc., encryption – decryption algorithms on symmetric key are very much popular in the Network Security System. In this paper, a numerical method based secret key encryption – decryption algorithm is developed using steganography. The idea of one way function [7] alongwith Newton's Method [8] is applied as a secret (symmetric) key and Digital Signature Standard (DSS) technology is used to send the key. Also, steganography is used to hide the cipher within a picture in encryption algorithm. Finally, the proposed encryption – decryption algorithm is discussed with a suitable conclusion.

The construction of the paper is as follows. In section 2, some prerequisite topics viz. Symmetric Key Cryptography, Steganography, DSS Technology, One-Way Function and Newton's Method are discussed. Then, in section 3, the proposed work along with the explanation of Algorithm is discussed. Finally, in section 4, some conclusions are specified.

2. PREREQUISITE TOPIC

Here, in this section, some prerequisite subject matters and mathematics are discussed and actually these topics are used to develop the paper.

2.1. Secret Key Cryptography or Symmetric Key Cryptography (SKC)

There are several ways of classifying cryptographic algorithms. These are categorized based on the number of keys that are employed for encryption and decryption, and further defined by application and use. Symmetric Key Cryptography (SKC) is one of these important parts.

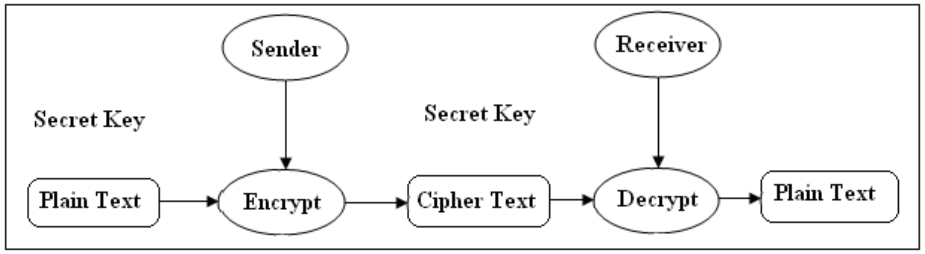

Figure 2: Symmetric Key Cryptography

Symmetric cryptography is also referred to as conventional encryption or single key encryption. In symmetric cryptography the same key is used for both encryption and decryption. This technique can encrypt data, either locally by a single user to safeguard his/her files, or to be exchanged between users. If encrypted data is exchanged between two (or more) users, each must know the key to be used. Obviously, this key should be exchanged in a secure manner. Symmetric cryptography is commonly used to perform encryption. It also provides data integrity when symmetric keys are used in conjunction with other algorithms.

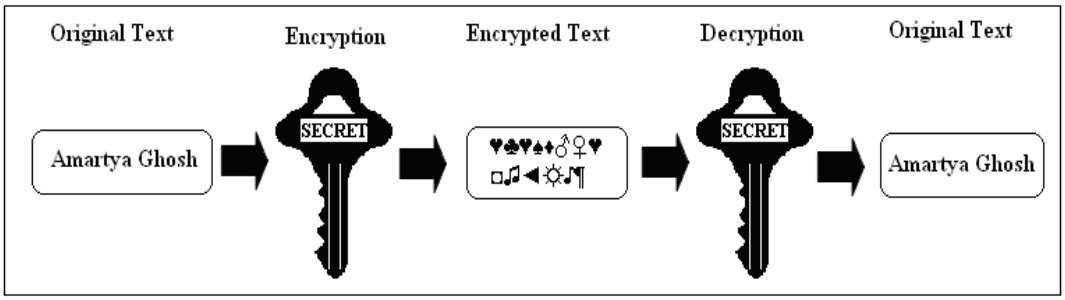

Figure 3: Symmetric Key Technique

2.2. Steganography

Steganography is the practice of hiding private or sensitive information within something that appears to be nothing out of the usual. Steganography is often confused with cryptology because these two are similar in the way that they both are used to protect important information. The difference between the two is that steganography involves hiding information so it appears that no information is hidden at all. If a person views the object in which the information is hidden and he or she have no idea that there is any hidden information, then the person will not attempt to decrypt the information. Steganography in the modern day sense usually refers to information or a file that has been concealed inside a digital Picture, Video or Audio file.

Steganography and encryption are both used to ensure data confidentiality. However the main difference between them is that with encryption anybody can see that both parties are communicating in secret. Steganography hides the existence of a secret message and in the best case nobody can see that both parties are communicating in secret. This makes steganography suitable for some tasks such as copyright marking. Adding encrypted copyright information to a file could be easy to remove but embedding it within the contents of the file itself can prevent it being easily identified and removed. Encryption allows secure communication requiring a key to read the information. Steganography provides a means of secret communication which cannot be removed without significantly altering the data in which it is embedded. The embedded data will be confidential unless an attacker can find a way to detect it. There are numerous methods used to hide information inside of Picture, Audio and Video files.

When hiding information inside images the LSB (Least Significant Byte) method is usually used. To a computer an image file is simply a file that shows different colors and intensities of light on different areas of an image. The best type of image file to hide information inside of is a 24 Bit BMP (Bitmap) image. The reason being this is the largest type of file and normally it is of the highest quality. When an image is of high quality and resolution, then it is a lot easier to hide and mask information inside of. Although 24 Bit images are best for hiding information inside of due to their size, some people may choose to use 8 Bit BMP's or possibly another image format such as GIF, the reason being is that posting of large images on the internet may arouse suspicion. It is important to remember that if you hide information inside of an image file and that file is converted to another image format, it is most likely the hidden information inside will be lost.
The art of detecting steganography is referred to as steganalysis. To put it simply steganalysis involves detecting the use of steganography inside of a file. Steganalysis does not deal with trying to decrypt the hidden information inside of a file, just discovering it. To detect steganography viewing the file and comparing it to another copy of the file found on the internet (Picture File). There are usually multiple copies of images on the internet, so you may want to look for several of them and try and compare the suspect file to them. For example if you download a JPEG and your suspect file is also a JPEG and the two files look almost identical apart from the fact that one is larger than the other, it is most probable your suspect file has hidden information inside of it.

2.3. Digital Signature Standard (DSS) Technology

A digital signature is an electronic analogue of a written signature. The digital signature can be used to provide assurance that the claimed signatory signed the information. In addition, a digital signature may be used to detect whether or not the information was modified after it was signed (i.e., to detect the integrity of the signed data). These assurances may be obtained whether the data was received in a transmission or retrieved from storage. A properly implemented digital signature algorithm that meets the requirements of this standard can provide these services.

2.4. One-Way Function

In computer science, a one-way function is a function that is easy to compute on every input, but hard to invert given the image of a random input. Here "easy" and "hard" are to be understood in the sense of computational complexity theory, specifically the theory of polynomial time problems. The existence of such one-way functions is still an open conjecture.

In applied contexts, the terms "easy" and "hard" are usually interpreted relative to some specific computing entity; typically "cheap enough for the legitimate users" and "prohibitively expensive for any malicious agents". One-way functions, in this sense, are fundamental tools for cryptography, personal identification, authentication, and other data security applications. While the existence of such functions too is an open conjecture, there are several candidates that have withstood decades of intense scrutiny. Some of them are essential ingredients of most telecommunications, e-commerce, and e-banking systems around the world.

This function must be "hard to invert" in the average-case. Note also that just making a function "lossy" (not one-to-one) does not make it a one-way function. In this context, inverting a function means identifying some pre-image element of a given value, which does not require the existence of an inverse function. For example, $f(x) = x^2$ is not invertible (for example $f(2) = f(-2) = 4$) but is also not one-way, since given any value, you can compute one of its pre-image elements in polynomial time by taking its square root.
If f is a one-way function, then the inversion of f would be a problem whose output is hard to compute (by definition) but easy to check (just by computing f on it). There is an explicit function which has been demonstrated to be one-way if and only if one-way functions exist. Since this

function was the first combinatorial complete one-way function to be demonstrated, it is known as the "universal one-way function". The problem of determining the existence of one-way functions is thus reduced to the problem of proving that this specific function is one-way.

2.5. Newton's Method

In numerical analysis, Newton's method (also known as the Newton–Raphson method), named after Isaac Newton and Joseph Raphson, is perhaps the best known method for finding successively better approximations to the zeroes (or roots) of a real-valued function. Newton's method can often converge remarkably quickly; especially if the iteration begins "sufficiently near" the desired root. Just how near "sufficiently near" needs to be, and just how quickly "remarkably quickly" can be, depends on the problem. This is discussed in detail below. Unfortunately, when iteration begins far from the desired root, Newton's method can easily lead an unwary user astray with little warning. Thus, good implementations of the method embed it in a routine that also detects and perhaps overcomes possible convergence failures.

Given a function $f(x)$ and its derivative $f'(x)$, we begin with a first guess x_0. Provided the function is reasonably well-behaved a better approximation x_1 is $x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$. The process is repeated until a sufficiently accurate value is reached i.e. $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$.

The idea of the method is as follows: one starts with an initial guess which is reasonably close to the true root, then the function is approximated by its tangent line (which can be computed using the tools of calculus), and one computes the *x*-intercept of this tangent line (which is easily done with elementary algebra). This *x*-intercept will typically be a better approximation to the function's root than the original guess, and the method can be iterated.

Suppose *f*: [a, b] → R is a differentiable function defined on the interval [a, b] with values in the real numbers R. The formula for converging on the root can be easily derived. Suppose we have some current approximation x_n. Then we can derive the formula for a better approximation, x_{n+1}. We know from the definition of the derivative at a given point that it is the slope of a tangent at that point.

So, $f'(x_n) = \frac{rise}{run} = \lim_{x_n \to x_{n+1}} \left(\frac{f(x_n) - 0}{x_n - x_{n+1}} \right)$, where, f' denotes the derivative of the function *f*.

Then by simple algebra we can derive $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$

- **Example for Square root of a number:** Consider the problem of finding the square root of a number. For example, if one wishes to find the square root of 612, this is equivalent to finding the solution for $x^2 = 612$. The function to use in Newton's method is then $f(x) = x^2 - 612$ with derivative $f'(x) = 2x$. With an initial guess of 10, the sequence given by Newton's method is

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = 10 - \frac{10^2 - 612}{2 \times 10} = 35.6$$

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} = 35.6 - \frac{35.6^2 - 612}{2 \times 35.6} = \underline{2}6.3955056$$

$$x_3 = x_2 - \frac{f(x_2)}{f'(x_2)} = \ldots\ldots\ldots\ldots\ldots\ldots = \underline{24}.\underline{7}906355$$

$$x_4 = x_3 - \frac{f(x_3)}{f'(x_3)} = \ldots\ldots\ldots\ldots\ldots\ldots = \underline{24}.\underline{738}\underline{6}883$$

$$x_5 = x_4 - \frac{f(x_4)}{f'(x_4)} = \ldots\ldots\ldots\ldots\ldots\ldots = \underline{24}.\underline{7386}\underline{33}\underline{8}$$

Where the correct digits are underlined. With only a few iterations one can obtain a solution accurate to many decimal places.

- **Example for solution of a non-polynomial equation:** Consider the problem of finding the positive number x with $\cos x = x^3$. We can rephrase that as finding the zero of $f(x) = \cos x - x^3$. We have $f'(x) = -\sin x - 3x^2$. Since $\cos x \le 1$ for all x, we know that solution lies between 0 and 1. We try a starting value of $x_0 = 0.5$. (Note that a starting value of 0 will lead to an undefined result, showing the importance of using a starting point that is close to the zero).

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = 0.5 - \frac{\cos(0.5) - (0.5)^3}{-\sin(0.5) - 3(0.5)^2} = 1.112141637097$$

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} = \ldots\ldots\ldots\ldots\ldots\ldots = \underline{0}.909672693736$$

$$x_3 = x_2 - \frac{f(x_2)}{f'(x_2)} = \ldots\ldots\ldots\ldots\ldots\ldots = 0.\underline{86}7263818209$$

$$x_4 = x_3 - \frac{f(x_3)}{f'(x_3)} = \ldots\ldots\ldots\ldots\ldots\ldots = 0.\underline{8654}\underline{7}7135298$$

$$x_5 = x_4 - \frac{f(x_4)}{f'(x_4)} = \ldots\ldots\ldots\ldots\ldots\ldots = 0.\underline{8654740}\underline{33}\underline{1}11$$

$$x_6 = x_5 - \frac{f(x_5)}{f'(x_5)} = \ldots\ldots\ldots\ldots\ldots\ldots = 0.\underline{8654740}\underline{33}\underline{102}$$

The correct digits are underlined in the above example. In particular, x_6 is correct to the number of decimal places given. We see that the number of correct digits after the decimal point increases from 2 (for x_3) to 5 and 10, illustrating the quadratic convergence.

3. PROPOSED WORK

In this section, the overall proposed work on numerical method based Encryption – Decryption Algorithm along with Steganography is discussed.

3.1. Proposed Algorithm

The Proposed Algorithms for Encryption and Decryption are given below:

- **Algorithm for Encryption:**

  Step–1: Read the characters from a text file and get the ASCII values for different characters.
  Step–2: Take a Polynomial function from Receiver using DSS Technology and subtract the ASCII values from the Polynomial and equating with zero to get the polynomial equations.
  Step–3: Solve the Polynomial Equations and put the solutions into an array.
  Step–4: Put the array into a BMP file.
  Step–5: Put some Garbage pixels into the BMP file.

- **Algorithm for Decryption:**

  Step–1: Read the BMP file and isolates the actual pixels from Garbage pixels.
  Step–2: Take the values from the pixels and store into an array.
  Step–3: Put the values into the Polynomial function and get the functional values.
  Step–4: Convert the functional values into characters and put into a text file.

3.2. Encryption Process (Sender Side)

1) Create a Text File: First of all we write a text message into a text file. Now using file pointer we read the characters and store the ASCII value of these characters.

2) Send the Request to the Receiver and get the One Way Function: Now send a request to the Receiver to give the One Way Function. Receiver will send the function using Digital Signature Standards (DSS). After receiving the function check the authentication of that file which is contained with the function. Receiver will send the function in a BMP file. Suppose the function is $f(x) = 3.5x^3 - 2.7x - 17$.

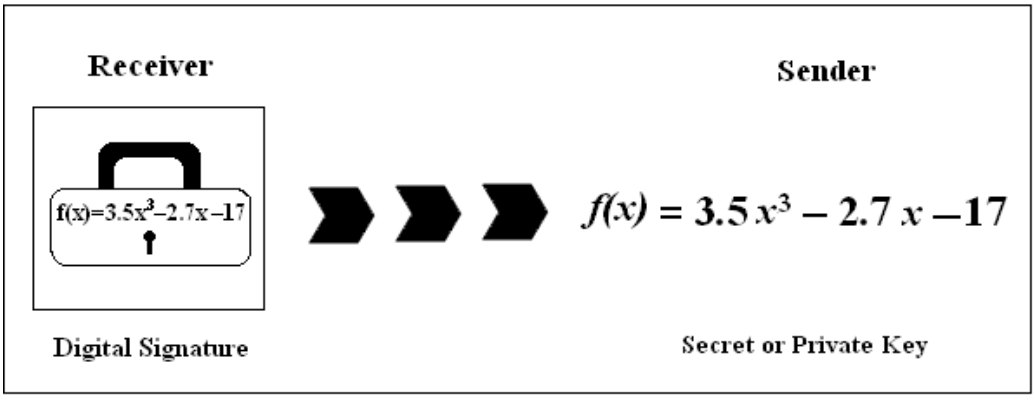

Figure – 4: Secret Key

3) Form the Equation using Function and Solve the Equation: Now for each ASCII value create an equation like $f(x)$ – ASCII Value = 0. Solve the equation and get the value of x. For the ASCII value of each character we get a solution and store it into memory. For example, ASCII value of M is 77 so that the equation will be $f(x)-77=0$. After solving the equation, the solution is $x=3.0805$. Similarly, solutions for other characters are also stored into memory.

4) Create the .BMP File: Now the solution is a Floating Point Number, which is required 4 bytes or 32 bits in the memory. Now split every 32 bit into four 8 bits. Each 8 bits will be printed as a pixel in a BMP file.

5) Put Garbage Pixels: Using some kind of sequence or a series, put some garbage pixel to make the encryption process more secure.

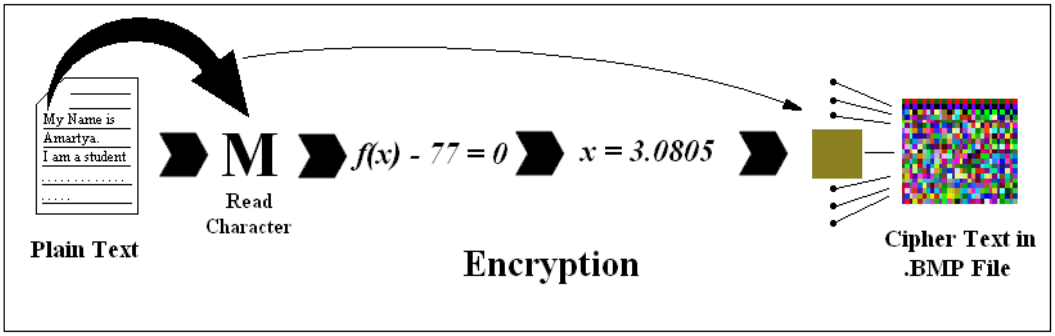

Figure 5: Encryption Process

3.3. Decryption Process (Receiver Side)

1) Read the .BMP File and isolate the original pixels: First of all read the BMP file and isolate the original pixels from the garbage pixels. From value of continuous four pixels we can get the each solution, which is a floating point number.

2) Calculate the Functional Values to get the ASCII Values of the Characters: Now put the solutions into the function and calculate the functional values. For example, f (3.0805) = 76.99585. Now rounding the result we can get 77, which is the ASCII value of 'M'. Finally write the character into a text file and then finding the other characters by similar way one can read and understand the total message.

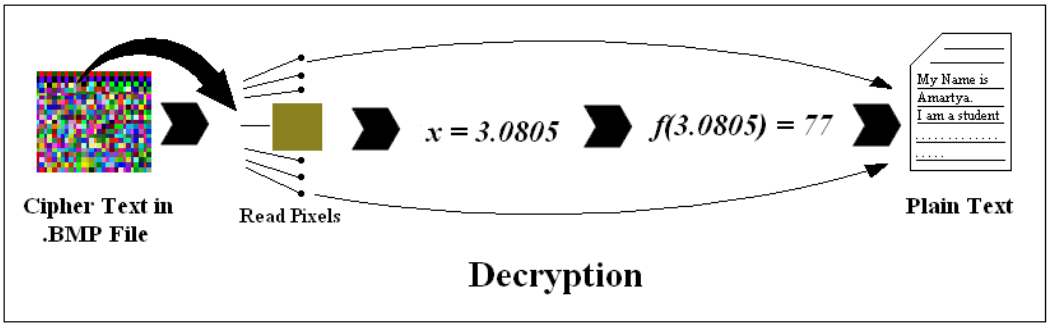

Figure – 6: Decryption Process

4. CONCLUSION

In this paper a new approach is proposed to secure the network by combination of cryptography and steganography. The proposed algorithm is very simple in nature, more secure and less complex and this algorithm is more useful for any kind of computer configuration. Here, a secret key conception is introduced taking the idea of one way function alongwith Newton's Method. Finally, a numerical method based secret key encryption – decryption algorithm is developed using steganography to enhance the Network Security System.

REFERENCES

[1] Stallings, W., (2002) "Cryptography & Network Security: Principals and Practice", 3rd Edition, Prentice Hall.
[2] Menzes, A. J., Paul, C., Van Dorschot, V., Vanstone, S. A., (2001) "Handbook of Applied Cryptography", CRS Press 5th Printing.
[3] Koblitz, N., (1994) "A Course in Number Theory and Cryptography", Springer-Verlag, New York, Inc.
[4] Shannon, C. E., (1949) "Communication Theory of Security System", Bell System Technical Journal, Vol. 28, No. 4, pp. 656 – 715.
[5] Ayushi, (2010) "A Symmetric Key Cryptographic Algorithm", International Journal of Computer Applications, Vol. 1, No. 15, pp. 0975 – 8887.
[6] Booth, Kellogg S., (1981) "Authentication of signatures using public key encryption", Communications of the ACM, pp. 772 – 774.
[7] Levin, Leonid A., (2003) "The Tale of One-way Functions", Problems of Information Transmission, Vol. 39, No. 1, pp. 92 – 103.
[8] Mollah, S. A., (2000) "Numerical Analysis and Computational Procedures", Books & Allied (P) Ltd. Kolkata.

AUTHORS

Amartya Ghosh received his B.Tech. and M.Tech. degrees in Information Technology and Computer Science Engineering respectively from JIS College of Engineering, Kalyani, West Bengal, India. He is currently an Assistant Professor at Regent Education & Research Foundation. His research interests include Cryptography, Financial Management and Soft Computing Technique.

Anirban Saha received his B.Sc. and M.Sc. degrees in Mathematics from the Dept. of Mathematics, University of Kalyani, West Bengal, India. Also he did his B.Ed. degree from the Dept of Education, University of Kalyani. He is currently an Assistant Professor (Senior Grade) at Regent Education & Research Foundation and pursuing Ph.D. degree in Mathematics at National Institute of Technology, Durgapur, India. He has some research publications in different reputed International and National Journals and Conferences. His research interests include Operations Research & Optimization, Financial Management and Soft Computing Technique.