ISSN:0975-9646

Defend Data using ELGAMAL Digital Signature Data Decryption Algorithm.

K Vahini[#1], V Prasad [*2] ,U V Chandra Sekhar[#3]

[#1]M.Tech Scholar, [#3]Assistant Professor ,Department of Computer Science & Engineering,
Raghu Engineering College, Dakamarri, ,Visakhapatnam-531162, Andhra Pradesh , INDIA.
*[*2]Associate Professor , Department of Computer Science & Engineering,*
Raghu Institute of Technology, Visakhapatnam, Andhra Pradesh, INDIA

Abstract*— Data security and encryption are now a days playing major role in Information Technology [IT] sector. Security problem generated creates exertion to the firm. They are several Digital Signature Algorithms which are useful in providing security for the issues generated. Elgamal Digital Signature [EDS] Algorithm which is used in wide applications had proved its efficiency in safe guarding the data .However due to different choppers the data is not firmly reaching the safe side. The previous methods proposed using this EDS Algorithm had given appropriate measures using several methods in protecting the data. But there are some flaws which made EDS Algorithm efficiency poor. In this paper, we are proposing an advanced EDS Algorithm with keys generated through statistical approach which consists of combination of random numbers and prime numbers blend with an Exclusive OR (⊕) operation to enhance the complexity for the key to be generated. We know that EDS Algorithm also ensures security and time complexity of improved signature. This proposed method can give us an authentication with a Digital Signature for decryption of the data at the receiver side very sanctuary.

***Keywords*— Data Security-Elgamal Digital Signature Algorithm-Data Encryption & Decryption-Sanctuary of Data-Random & Prime Numbers-Exclusive OR-Sender –Reliever.**

I INTRODUCTION

Network Security & Cryptography [NSC][2] is to protect network and data transmission over wireless network. Data Security is to secure data transmission over unreliable network. The rapid development in IT, the secure transmission of confidential data herewith gets a great deal of attention. The conventional methods of encryption can only maintain the data security. The information could be accessed by the chopper for malicious purpose. Therefore, it is necessary to apply an effective encryption/decryption algorithms to enhance Data Security [DS][1].

A **Digital Signature [DIS]** is a mathematical scheme for demonstrating the authenticity of a digital message or document. The valid DIS[5] gives a reason for recipient to believe that the message was created by a known sender such that they cannot deny sending it, so that the message was not altered in transit. This DIS concept creates Message Digest [10][MD] which identifies a message uniquely.

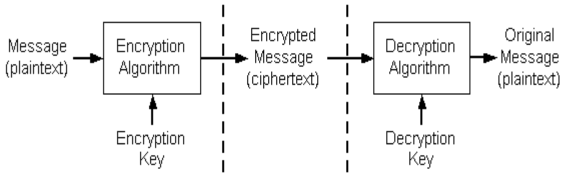


Fig 1.1 General method of Cryptography

They are originally two types of cryptographic process ***namely (1)*** *Enciphe**r** process to convert plain text to cipher text.(2) Decipher process to convert cipher text to plain text.and similarly the text types are* **(i)Plain text**: *which* is used as input to an encryption algorithm; the output is usually termed as cipher text.**(ii)Cipher text:** *Cipher text* is also known as encrypted or encoded information because it contains a form of the original plain text.

Keys [12] the major aspect in NSC is piece of information (a parameter) that determines the i.e. a Functional output of a cryptographic algorithm. And the key types are

(i) Private Key*:* is a secret key and also known as encryption/decryption key known only to the party or parties that exchange private messages.

(ii) Public Key*:* is created in encryption cryptography that uses asymmetric-key encryption algorithms. Public keys are used to convert a message into an unreadable format.

ELGAMAL DIGITAL SIGNATURE ALGORITHM

The difference between the proposed algorithm and the original EDS[1] algorithm is mainly reflected in adding a random number along with Exclusive-OR operations aiming to make the original algorithm more complicated and more difficult to decipher[1][3].

The specific algorithm is as follow:

STEP 1: A large prime number p is produced by system, α is a generator of Zp*, x(1< x <□(p)) is the signer's private key, the corresponding signature public key β can be calculated by$\beta = \alpha\ x \bmod p$, and opened to the public key.

STEP 2: Two different random numbers t and k are randomly selected by system. Where t, k and x must be co-prime and there is inverse. γ and λ are calculated by the $\gamma = \alpha\ k \bmod p$, $\lambda = \alpha\ t \bmod p$ and retain γ and λ.

STEP 3: Signature explicitly m, δ is calculated using the results of the first two steps as well as the extended Euclidean algorithm and modular inversion algorithm by **$m = (x\gamma + k\lambda + t\delta)$ mod (p −1)** It should be avoided to take the same random number and simple functional relationship existing between random numbers at the course of obtaining a number of signatures. [3]

STEP 4: Discarded the random number k and t, then the required public key p, β and α are obtained. The private key is x. The signature of plain text m is (γ, λ, δ)

STEP 5: (γ, λ, δ) is sent to the corresponding customers by system. The customers use the following equation to verify the correctness of plaintext m digital signatures. If equal, the signature is correct. Otherwise, the signature is incorrect or transmission errors. The equation as follows:

$\alpha\ m = \beta\gamma\gamma\ \lambda\lambda\delta$ mod p

In the above-mentioned improved EDS algorithm, the same message m corresponded to the different digital signature (γ ,λ,δ) for the different random number k, t. And they can be all verified through the validation algorithm, which characterizes with uncertainty of signature and improved security. When signers take λ as a signature, they need to finish one more computing power each time, which increases the amount of the signer's operations. When taken λ as a public key, the signer calculates a value of λ, which could be used as many times as the value of β. So the signer's computation amount is almost the same as that of the original algorithm. But for authenticator, each authentication has one more computing power, but increased with only about 0.5 time computation. As for the computer which can easily verify the Elgamal-type digital signature, verifying the signature of the improved algorithm does not consume more time. λ can be used as public key if it is taken into account that the amount of the signer's operation is not increased[5]. But whether this is safe, it is still needed to be determined through the analysis of safety. If the public key λ as a random number will lead to insecurity, λ should be taken as a part of the signature. This would increase the operation of the signer. But as long as the signature is not carried out in large amount, this computation can still be accepted [7].

- security depends on the difficulty of computing discrete logarithms
- setup (key generation) is the same:
- shared prime p, public primitive root a
- each user chooses as private (key) a random number x
- and computes public key: y = ax mod p
- public key is (y,a,p)
- private key is (x)

To **sign** message M:

- choose a random number k, GCD(k,p-1)=1
- compute K = ak(mod p)
- use extended Euclidean (inverse) algorithm to find S:
  M = x.K + k.S mod (p-1); that is find
  S = k-1(M - x.K) mod (p-1)
- the signature is (K,S)

Note that k should be destroyed after use like Elgamal encryption the signature is double the message size to verify a signature (K,S) on message M confirm that:

yK.KSmod p = aMmod p

II PROBLEM SPECIFICATION:

Elgamal proposed a cryptosystem based on the Discrete Logarithm [DL] problem [13] which can be used for both data encryption and digital signature. The EDS scheme is non deterministic like Elgamal public-key cryptosystem [12]. The same clear message has different signatures due to randomly selected different parameters, and each signature only corresponds to a random number, which has brought a great hidden danger to the security of EDS scheme. In terms of the EDS scheme, the algorithm's security depends on the security of the private key. Once the private key [9-12] is intercepted by the chopper, the entire digital signature algorithm is accessible to all chopper can easily use the link among random numbers, and can decipher the private key as a primary target and obtain the value of the private key by no means of complicated calculations. Since the selection of random numbers is greatly reduced, and each random number can only be used once and then discarded, which has seriously affected the life of the algorithm.

The replacement of the public key needed to be re-issued through public channels. Therefore, some customers failed to access the updated information and still used in consistent public key for authentication leading to the erroneous conclusion. The error probability will increase resulting from over loaded data transmission. So, there exists more digital signature retransmission due to the error in transfer process. Thus, in the EDS, the insecure random number constitutes a very large threat to its security. To improve the algorithm's security, it is important that algorithm by adding a random number and strengthening the link between the random number and the private key to make it more difficult to decipher, besides this we also propose a method by adding a prime number along with the random number with an Exclusive-OR operation, which leads to high security in transferring the data, where the private key cannot be hacked so easily.

III PROPOSED SYSTEM:

According to the two method of analysis to attack the random number, it was found that the chopper can easily calculate the value of random numbers or the value of the key by calculation of a random number, if a signer uses the insecure random number. This generally resulted from that it is easier to hack the random number than hack the key or too intimate relationship between the random number and the key. So, for the random number vulnerability in the EDS algorithm and too simple link between the random number and the key.

In this paper, we proposed to enhance the security of the algorithm, which can make the link between the random number and the key more complicated.

There are two signature equations of the EDS algorithm, shown as follow:[1][3][5][7]

$\gamma= \alpha k \bmod p$ ---------------------------- (1)

$\delta= (m-x\gamma)k-1 \bmod (p-1)$ ----------------(2)

Public key β is calculated by the following equations:

$\beta= \alpha x \bmod p$ ---------------------------- (3)

Compared with (1), (3) is the same as (1) in form, with β generated through the private key x and as a public key. In (1) γ is generated through the random number k and as a part of the signature. Then we can join these three equations as a signature equation, but calculated as a public key instead of as a signature.

If we take β as part of the signature, γ as a public key, corresponding to x as a random number, k as the private key, the equation (2) is changed to follow:

$\delta= (m-x\gamma)k-1 \bmod (p-1)$ ----------------(4)

Verification equation is changed as follow:

$\alpha m = \beta\ \gamma\gamma\lambda\ \lambda\delta \bmod p$ -------------------(5)

This result of replacement is also an ElGamal digital signature algorithm. It can be seen that there is no essential difference between the random number k and the private key x. They are in different positions only because (2) is different. Then we can consider adding such a random number, and a corresponding increase in a form such as the type (3) of the equation to the signature equation, namely:

$\lambda = \alpha t \bmod p$ -------------------(6)

Accordingly, its need to make the appropriate changes to the signature equation and the verify equation. The signature equation in (2) is changed as follow:

$m = (x\gamma + k\lambda + t\delta) \bmod (p - 1)$-------------(7)

The authentication equation is changed as follow:

$\alpha m = \beta\ \gamma\gamma\lambda\ \lambda\delta \bmod p$-------------(8)

In this way, the linkage between the random number and the private key x is established based on (7). For (7), the values of x, k and t are to be identified. If the hackers successfully obtained a random number value and want to continue hacking, this is clearly more difficult than that for the original algorithm. Further improved the scheme above, the new digital signature

algorithm's signature equation can be obtained as follow:

$$\beta= \alpha x \bmod p$$
$$\gamma = \alpha k \bmod p$$
$$\lambda = \alpha t \bmod p$$
$$m = (x\gamma + k\lambda + t\delta) \bmod (p - 1)$$

Verification equation can be drawn as follows:

$$\alpha m = \beta\ \gamma\gamma\lambda\ \lambda\delta \bmod p$$

IV REQUIREMENT ELICITATION:

Domain Analysis:

Network Security and Cryptography

Network security consists of the provisions and policies adopted by a network administrator to prevent and monitor unauthorized access, misuse, modification, or denial of a computer network and network-accessible resources. Cryptography includes techniques such as microdots, merging words with images, and other ways to hide information in storage or transit.

Modern cryptography concerns itself with the following four objectives:

(i) **Confidentiality:** The information cannot be understood by anyone for whom it was unintended.

(ii) **Integrity:** The information cannot be altered in storage or transit between sender and intended receiver without the alteration being detected.

(iii) **Non-repudiation:** The creator/sender of the information cannot deny at a later stage his or her intentions in the creation or transmission of the information.

(iv) **Authentication:** The sender and receiver can confirm each other's identity and the origin destination of the information Cryptosystems are often thought to refer only to mathematical procedures and computer [9]. programs; however, they also include the regulation of human behaviour, such as choosing hard-to-guess passwords, logging off unused systems, and not discussing sensitive procedures with outsiders.

Software Requirements Specification:

A Software Requirements Specification (SRS) is a complete description of the behaviour of the system to be developed. It includes a set of use cases that describe all the interactions of the users will have with the software.

Functional Requirements

Functional requirements are supported by non-functional requirements (also known as quality requirements), which impose constraints on the design or implementation (such as performance requirements, security, or reliability).

- **Inputs:**
  - ✓ *Login details:* Users chose an id and password and validate their details.
  - ✓ *Key generation :* private key and public key of both the users are generated.
  - ✓ *Plain text* :sender selects or enters some text that is to be sent to the receiver.
- **Outputs:**
  - ✓ *Cipher text :* The encrypted text which is of form of original plain text viewed by receiver.
  - ✓ *Signature comparison :* verifies the authentication of the user and message contents.

Non Functional Requirements:

In systems engineering and requirements engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours. Other terms for non functional requirements are "constraints", "quality attributes", "quality goals" and "quality of service requirements," and "non-behavioural requirements." Qualities, that is, non-functional requirements, can be divided into two main categories:

1. Execution qualities, such as security and usability, which are observable at run time.

2. Evolution qualities, such as extensibility, scalability and reliability which are embodied in the static structure of the software system.

Execution Qualities:

Security:

The application being developed has a login page which provides security to the application. Only the authenticated person can access it , as it is verified through digital signature.

Usability:

The application will have a very simple user interface. So a person who is operating the application for the first time will be able to use it.

Evolution Qualities:

Extensibility:

We will be developing the project with a scope of extending it anytime in the networking environment**.**

Scalability:

The application being developed is a standalone application. Hence it could be installed in any number of systems and could be used easily.

Reliability:

The system is intended to be developed with minimal errors, so that it doesn't deviate from its actual Behaviour.

Hardware Requirements

1. VDU: Monitor/ LCD TFT / Projector
2. Input Devices: Keyboard and Mouse
3. RAM: 512 MB
4. Processor: P4 or above
5. Storage: Less than 100 MB of HDD space.

Software Requirements:

1. Operating System: Windows XP SP3 or above
2. Run-Time: .Net Framework 4.0 or above

V.MODULES:

System implementation:

The proposed system is implemented to the best of no deviations. The implemented part has the below modules.

1. Authentication
2. Choosing or generating random number
3. Encryption/Decryption
4. Signature Generation
5. Comparison

Authentication:

The login screen is displayed as soon as the executable file runs. The login screen is expected to provide a valid user name and password on failing which the rest of the application is not allowed to be accessed.

Generating keys: The user generates the keys required for transaction.

Encryption/Decryption: The user encrypts and then decrypts the data based on the keys

Signature Generation: Vulnerable Data consists of 0's and 1's are displayed in a equi format.

Comparison: Later the signature is calculated on the data and then compared.

VI. USER INTERFACE DIAGRAM:

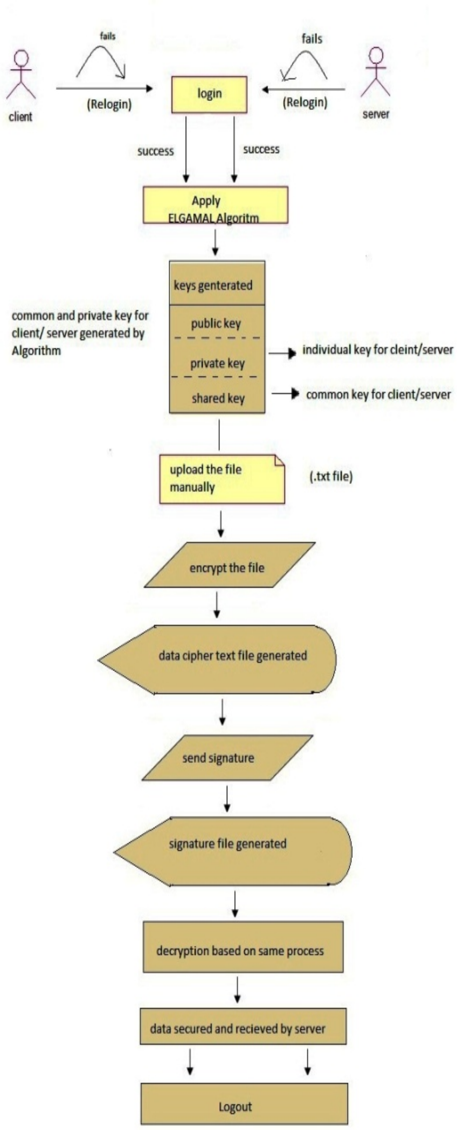
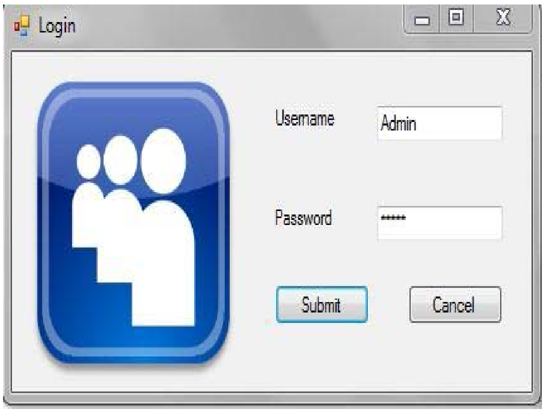

fails
fails
client
(Relogin)
login
(Relogin)
server
success
success
Apply
ELGAMAL Algoritm
keys genterated
common and private key for client/ server generated by Algorithm
public key
private key
shared key
individual key for cleint/server
common key for client/server
upload the file manually
(.txt file)
encrypt the file
data cipher text file generated
send signature
signature file generated
decryption based on same process
data secured and recieved by server
Logout

Fig6.1. Illustrating the method of proposed work

VII.RESULTS:

SERVER SIDE:

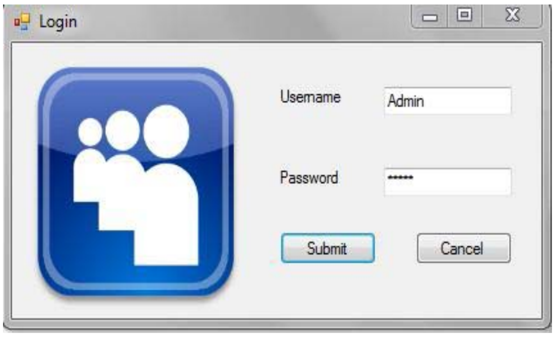


Username Admin

Password *****

Fig 7.1: Basic screen shot for server page

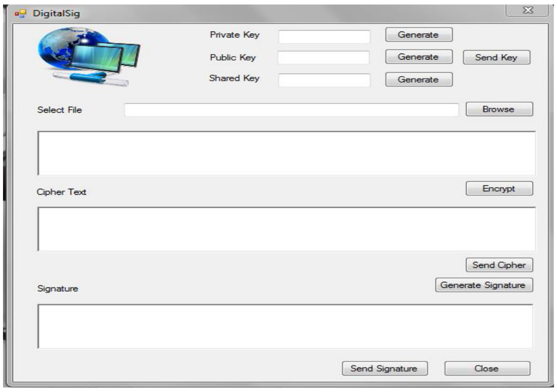
CLIENT SIDE:


Login

Username: Admin

Password: *****

Fig 7.2: Basic screen shot for client page


DigitalSig

Private Key

Public Key

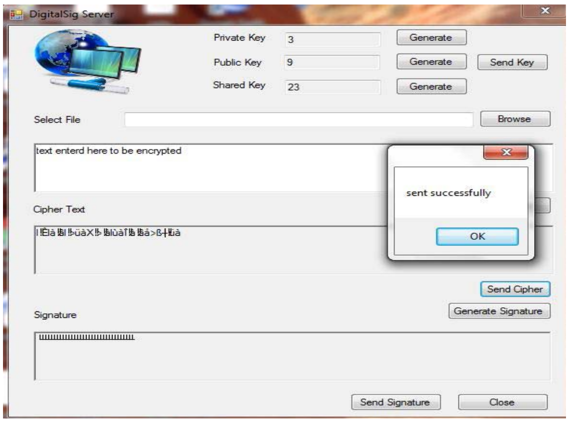
Shared Key

Select File

Cipher Text

Signature

Fig 7.3:Form on server slide with keys and digital signature ready for evaluation


DigitalSig Server

Private Key: 3

Public Key: 9

Shared Key: 23

Select File

text enterd here to be encrypted

Cipher Text

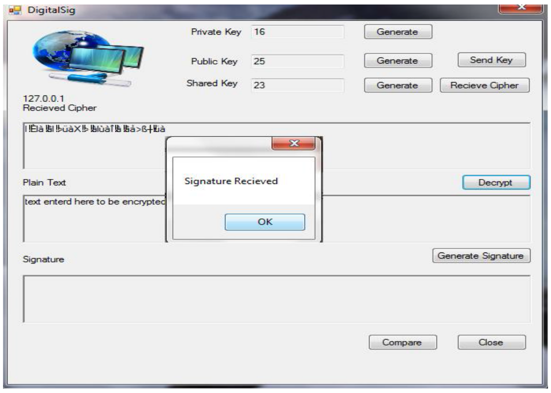
‼Èà ‼ ‼üàX‼ ‼ùàî‼ ‼å>ß‡‼à

Signature

ııııııııııııııııııııııııııııııııı

sent successfully

Fig 7.4: Successfull generation of keys on server slide and sent to client


DigitalSig

Private Key: 16

Public Key: 25

Shared Key: 23

127.0.0.1

Recieved Cipher

‼Èà ‼ ‼üàX‼ ‼ùàî‼ ‼å>ß‡‼à

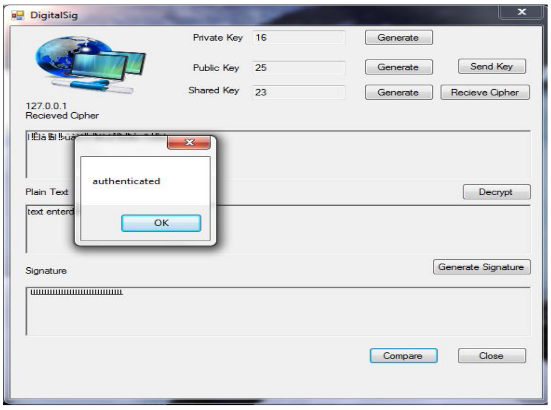
Plain Text

text enterd here to be encrypted

Signature Recieved

Signature

Fig 7.5: Successfull generation signature received


DigitalSig

Private Key: 16

Public Key: 25

Shared Key: 23

127.0.0.1

Recieved Cipher

‼Èà ‼ ‼üà

Plain Text

text enterd

authenticated

Signature

ııııııııııııııııııııııııııııııııı

Fig 7.6: Authentication is successful and complete transfer of data from client to server proposing a digital signature

CONCLUSION:

In order to enhance the security of algorithm in random, we proposed two improved ideas: enhanced security of random numbers, making it difficult for the success of the random number of hacks; establish more complex link between the random number and the private key, so it is difficult for a chopper to use random number to indirectly attack the private key, based on ideas that established the more complex link between the random number and the private key, we proposed to add the signature equation of the same form of an equation with the improvement of the program by Exclusive-OR (⊕) operation which minimized by the chopping level.

APPENDIX

USER MANUAL

The project **"Defend Data using ELGAMAL Digital Signature Data Decryption Algorithm."** standalone application developed using Dot Net Framework 4.0. The following are the instructions that are to be followed in order to deploy the software.

The following are the requirements for successful deployment.

1. Dot Net Framework 4.0
2. Complete Well Running ELGAMAL Algorithm
3. Linux/ Windows 98 or above

ANY JAVA ENABLED WEB BROWSER.

ACKNOWLEDGMENT

Apart from my individual effort, the success of the project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my gratitude to all of them and especially our Head of the Department , **Professor Sivakoti Satyanarayana**.

REFERENCES

[1] An emproved Elgamal Digital Signature Algorithm Based on adding a Random Number, 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing.

[2] Cryptography and Network Security – Atul Kahate,Tata McGraw

[3] WANG Hua-qun, XU Ming-hai, GUO Xian-jiu. Cryptanalysis and improvement of several certificateless digital signature schemes[J]. Journal on Communications, 2008, 28(5):88-92. (in chinese)

[4] Network Security Private Communication in a public world- Charlie Kaufman, Radi Perlman

[5] RAFAEL C, RICARDO D. Two notes on the security of certificateless signature[A]. Provsec 2007[C]. Springer- Verlag, 2007. 85-102.

[6] Network security Essentials Applications and standards-William Stallings

[7] LI Wei-ke, LI Fang-wei.User authentication scheme based on the ElGamal signature for mobile communication system[J]. Journal on Communications, 2005, 26(11): 138-140. (in chinese)

[8] Claude Castelluccia ,Nitesh Saxena ,Jeong Hyun Yi. Self-configurable key pre-distribution in mobile Ad Hoc Networks[J] .Lecture Notes in Computer Science. 2005, 1083 - 1095.

[9] WANG Shu-hong, WANG Gui-lin, BAO Feng, et al. Cryptananlysis of a Proxy Blind Signature Scheme Based on DLP[J]. Journal of Software, 2005,16(5): 911-915. (in chinese)

[10] Complete Reference: Network security -Tanenbaum

[11] DONG Qing-Kuan, NIU Zhi-Hua, XIAO Guo-zhen. Research on the Freeness of Subliminal Channels in ElGamal-Type Signatures[J]. Chinese Journal of Computers, 2004, 27(6):845-848. (in chinese)

[12] CAO Zhen-fu, LI Ji-guo. A Threshold Key Escrow Scheme Based on EiGamal Public Key Cryp to system[J]. Chinese Journal of Computers, 2002,25(4):346-348. (in chinese)

[13] ELGAMAL T. A public key cryptosystem and a signature scheme based on discrete logarithms[J]. IEEE Trans Inform Theory.1985,31(4): 469- 472.

AUTHORS:

Ms K Vahini , present she is pursuing M.Tech in Raghu Engineering College . She completed her B.Tech in Pydah College of Engineering & Technology in 2007. She have interests on the subject like Computer Networks , Data Communications & Cryptanalysis.. Attended workshops and Seminars on Cryptography and Network Security.

Mr U V Chandra Sekhar , working as an Assistant Professor in Raghu Engineering College since 7 Years and fetching a total experience of 10 Years . He had good expert knowledge in Networks , Micro Processors & Databases and had rich journals published by him. He had guided many UG and PG Projects which are used in college / small firm levels also.He completed his Msc in 2004 and M.Tech in 2009 and the certifications are from Andhra University . A Good and Kind Hearted person .

Mr V Prasad , working as an Associate Professor in Raghu Institute of Technology since 8 Years having enough knowledge in Algorithms related to Machine Learning , Expert Systems & Artificial Intelligence . He is awarded Bachelors Degree(Computers) from JNTU-H , Masters Degree(CST – AI & Robotics) from Andhra University and currently pursuing Ph.D(Computers) in Gitam University. He had several publication in National / International Journals, and attended many n National Seminars. Presented papers in International conferences.