Message Encryption Using Deceptive Text and Randomized Hashing

VAMSIKRISHNA YENIKAPATI
Student, (M.Tech)
Department of Computer Science&Engineering, Chaitanya Engineering College
Visakhapatnam, Andhra Pradesh, India
E-mail: vamsia5@yahoo.co.in

B.POORNA SATYANARAYANA
Associate Professor
Department of Computer Science&Engineering, Chaitanya Engineering College
Visakhapatnam, Andhra Pradesh, India
E-mail: poornasatyanarayana@gmail.com

Abstract-- In this paper a new approach for message encryption using the concept called deceptive text is proposed.In this scheme we don't need send encrypted plain text to receiver, instead, we send a meaningful deceptive text and an encrypted special index file to message receiver.The original message is embedded in the meaningful deceptive text.The positions of the characters of the plain text in the deceptive text are stored in the index file.The receiver decrypts the index file and gets back the original message from the received deceptive text. Authentication is achieved by verifying the hash value of the plaintext created by the Message Digest Algorithm at the receiver side.In order to prevent collision attcks on hashing algorithms that are intended for use with standard digital signature algorithms we provide an extra layer of security using randomized hashing method.

Keywords: AES, Character Position Table (CPT), Deceptive Text, MD5, Plain Text Index File (PIF), plaintext, Randomized Hashing

I. INTRODUCTION

In the past,most researchers in the area of data security have put a great effort on developing cryptosystems which are difficult to reverse from ciphertext into the original data.But the ciphertext form will become meaningless after the original data have been encrypted.When a hacker gets these data,meaningless words,he definitely know that these data has been encrypted.Then if he want,he will try to decrypt these data.In other words,as soon as he gets these encrypted data,he exactly knows there is some important message he has obtained.

In this paper we propose that a deceptive text intead of encrypted plaintext is sent to receivers.The deceptive text can use any kind of text form which is constructed by ASCII code.The deceptive text can have no relation with the original plaintext.But, the deceptive text has a restriction: its number of character's kinds can't less less than the number of character's kinds used by the plaintext.

For example,if the plaintext is "I love India".The kinds of ASCII code's characters are 'I','space','l','o','v','e','n','d','i','a',totally 10 kinds of characters.And the deceptive text must use some words at least including 10 kinds of ASCII code's characters.We can write a deceptive text such as "I have played guitar for a long time".Then according to the plaintext and depending on the deceptive text we can make a special index file,called plaintext index file(PIF).

In this scheme we have a Character Position Table(CPT),which contains positions of each character in the deceptive text.From the CPT we generate Plaintext Index File(PIF),which is a series of character positions taken from CPT that correspond to the characters in the plaintext. Finally we compress and encrypt the Plaintext Index File (PIF) and send it to the receiver.Therefore, the real data we send is the deceptive text and the encrypted index file.

At the receiver the PIF is decrypted and the CPT is generated from the cheating text.The plaintext is generated by indexing PIF into the CPT.In this scheme we can also use digital sinatures to provide authentication for the receiver about the Deceptive text.Digital signature algorithms take hash value of the message (message for which we have to generate digital signature, in this case deceptive text) as input.A hash function 'H' is said have "collision" if H(x) =H(y) for x not equal to y.A good hash function is one is the one

which is collision free.If a given value x,it is computationally infeasible to find a value y(not equal to x) such that H(x)=H(y) then H is said to be a collision-free hash function.Normally hash functions suffer from various types of collisions.In order to free hash functions from these collisions we use a method called "randomized hashing".In randomized hashing we apply randomization to the message before applying it to the hash function.

II. RANDOMIZED HASHING

Randomized hashing is a method to enhance the security of the cryptographic hash functions used in digital signature applications by randomizing the messages that are to be signed.When using digital signature algorithms to generate a digital signature for a message, the message must first be processed using one of the approved hashing algorithms.We apply the randomization to the message prior to hashing.

Algorithm:-

Input:-

rv: random bit string.
Ms: message to be randomized.

Output:-

M: a randomized message.

Message Randomization (Ms, rv)
{
1. If ($|Ms| \geq (|rv|-1)$)
  {
    Padding=1
  }
 Else
 {
   Padding= $1 \| 0^{(|Ms|-|rv|-1)}$
 }
2. m=Ms$\|$padding
3. n is a positive integer, and n=$|rv|$
4. If (n>1024) then stop and output an error
5. counter=$|m| / n$
6. remainder=($|m|$ mod n)
7.Concatenate counter copies of the rv to the remainder left-most bits of the rv to get Rv,such that $|Rv|=|m|$
8. Convert n to a 16-bit binary string rv_length_indicator
9. M=rv$\|(m \oplus Rv)\|$rv_length_indicator
}

Output:-M

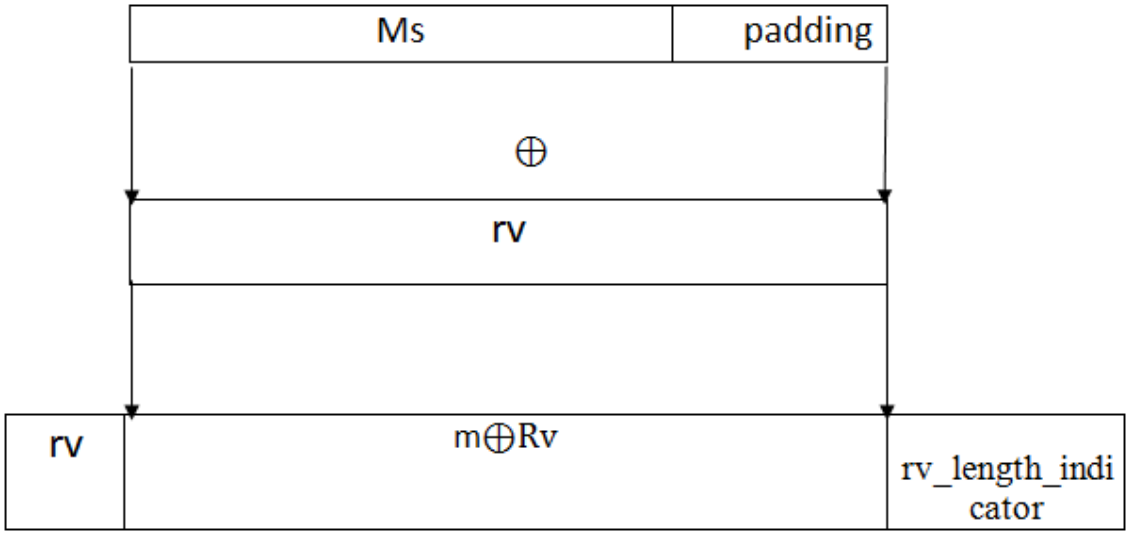

Fig 1: Process of randomizing a message

In this scheme, the security of the plaintext index file is very important.And we use AES (Advanced Encryption Standard) method to encrypt the index file.

III. OVERVIEW OF MD5

The following are the steps that are to be followed for generating hash value using MD5:

1) **Data Padding:-**The message is padded sothat its length,in bits,is 64 bits short of being a multiple of 512 bits long.
2) **Append length:** - A 64-bit representation of the original message length is appended to the result of the previous step.
3) **MD buffer Initialization:** - A four-word buffer, called MD buffer, is used to compute the message digest.This buffer is initialised with fixed hexadecimal values.
4) Process the message in 512-bit blocks.

The resulting hash value of MD5 algorithm will be provided as input to a digital signature algorithm to generate digital signature of message.

IV. ADVANCED ENCRYPTION STANDARD

AES is symmetric block cipher.A block cipher processes the plaintext input in fixed-size blocks and produces a block of ciphertext of equal size for each plaintext block.AES uses a block length of 128 bits and a key length that can be 128, 192, or 256 bits.

The following points give some insight into AES:

i. The key that is provided as input is expanded into an array of forty-four 32-bit words, w[i].Four distinct words serve as a round key for each round.
ii. Four different stages are used,one of permutation and three of substitution:
   - **Substitute bytes**: Uses a table, referred to as an S-box, to perform a byte-by-byte substitution of the block.
   - **Shift rows**: A simple permutation that is performed row by row.
   - **Mix columns**: A substitution that alters each byte in a column as a function of all the bytes in the column.
   - **Add round key**: A simple bitwise XOR of the current block with a portion of the expanded key.
iii. For both encryption and decryption, the cipher begins with an Add Round Key stage, followed by nine rounds that each includes all four stages, followed by a tenth round of three stages.

iv. Only the Add Round Key stage makes use of the key.For this reason, the cipher begins and ends with an Add Round Key stage.Any other stages, applied at the beginning or end, is reversible without knowledge of the key and so would add no security.

v. The Add Round Key stage itself is formidable.The other three rounds together scramble the bits, but by themselves would provide no security because they do not use the key.We can view the cipher as altenating operations of XOR encryption (Add Round Key) of a block, followed by scrambling of the block (the other three stages), followed by XOR encryption, and so on.This scheme is both efficient and highly secure.

vi. Each stage is easily reversible.For the Substitute Byte,Shift Row,and Mix Columns stages,an inverse function is used in the decryption algorithm.For the Add Round Key stage,the inverse is achieved by XORing the same round key to the block,using the result that A⊕A⊕B=B.

vii. As with most block ciphers, the decryption algorithm makes use of the expanded key in reverse order.However, the decryption algorithm is not identical to the encryption algorithm.

viii. The final round of both encryption and decryption consists of only three stages.

V. ENCRYPTION PROCESS

Algorithm:-

Input:- Plain Text, Deceptive Text.

Procedure:-

1. Verify whether character's kinds in deceptive text satisfy character's kinds in plain text.If satisfied, goto step2.If not, discard the current deceptive text and generate new one.
2. Generate Character Position Table (CPT) from the deceptive text.
3. Generate Plaintext Index File (PIF) using CPT.
4. Randomize deceptive text using a random value (rv).
5. Generate digital signature (DS) of randomized deceptive text using MD5 algorithm.
6. Encrypt PIF and rv using AES.
7. Send deceptive text, encrypted PIF, encrypted rv, and digital signature (DS).

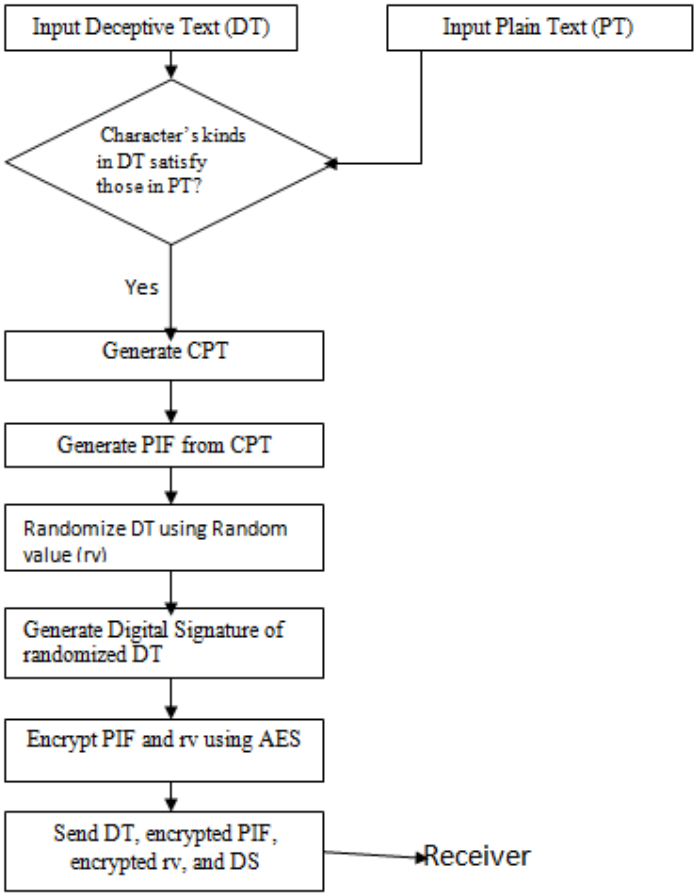

Fig 2: Process of Encryption

VI. DECRYPTION PROCESS

Algorithm:-

Input:-Digital signature (DS), Encrypted Random value (rv), Deceptive Text (DT), Encrypted Plaintext Index File (PIF).

Output:-Plaintext.

Procedure:-

1. Decrypt Random value (rv) using symmetric AES.
2. Randomize Deceptive Text (DT) using Random value (rv).
3. Generate digital signature (DS^1) of randomized deceptive text.
4. Verify whether received digital sinature (DS^1) are DS are equal.If equal goto step5.If not, discard received DS.
5. Decrypt PIF using symmetric AES algorithm.
6. Generate Character Position Table (CPT) from deceptive text.
7. Recover plaintext by indexing PIF into CPT.

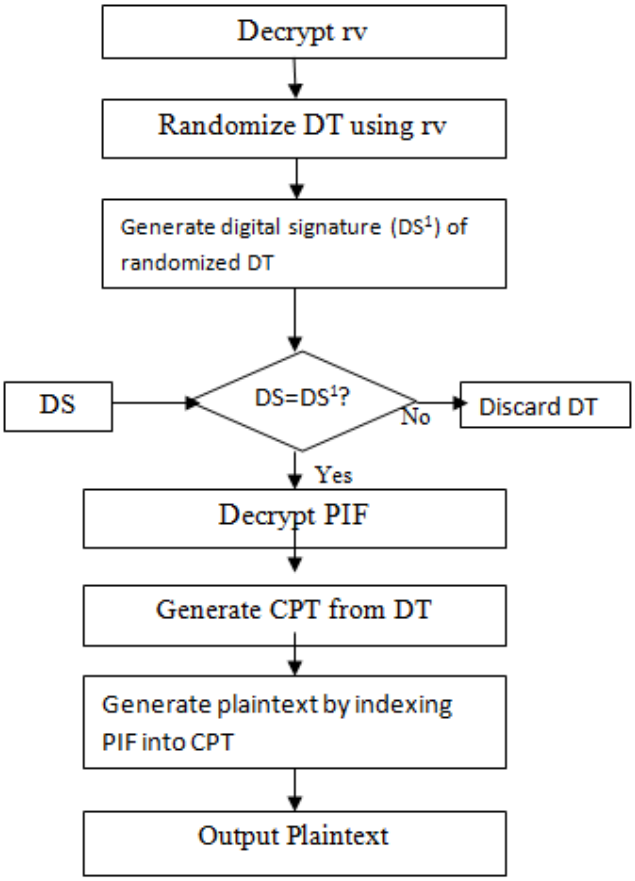

Fig 3:Process of Decryption

VII. EXAMPLE

We assume that

Plaintext: cat is my pet.

Cheating text: computer security is important.

According to the cheating text, we generate the CPT as follows:

Character	Position Record
C	1
o	2,25
m	3,23
p	4,24
u	5,13
t	6,16,27,30
e	7,11
r	8,14,26
space	9,18,21
s	10,20
i	15,19,22
y	17
a	28
n	29
.	31

Fig 4: Character Position Table

Then we compare each character in plaintext with CPT's characters and randomly chose a position record to make the PIF.
PIF is 1 28 16 21 15 20 18 3 17 9 24 7 6 31

VIII. CONCLUSION

Message encryption scheme using deceptive text and randomized hashing is proposed.This scheme reduces computational complexity of encrypting entire plaintext by encrypting just Plaintext Index File(PIF).And also it overcomes the "collision" weakness of hashing algorithms by using randomized hashing method.Finally,it provides a high-level of protection for the data at low computational complexity.

REFERENCES

[1] Xiaoyun Wang and Hongbo Yu,"**How to Break MD5 and other Hash Functions**".
[2] I.N.Tselepis, M.P.Bekakos, A.S.Nikitakis and E.A.Lipitakis,"**MD5 Hash Algorithm Hardware Realization on a reconfigurable FGPA Algorithm**".
[3] Shai Halevi and Hugo Krawczyk,"**Strengthening Digital Signatures via Randomized Hashing**".
[4] Quynh Dang,"**Randomized Hashing for Digital Signatures**".
[5] Chu-Hsing Lin and Tien-Chi Lee,"**A Confused Document Encryption Scheme and Its Implementation**".
[6] William Stallings,"**Network Security Essentials**".
[7] Brian Murray,"**An analysis of the MD5 hashing algorithm**".
[8] "**The Digital Signature Standard Proposed by NIST**"

AUTHORS PROFILE

VamsiKrishna yenikapati is M.Tech student at Chaitanya Engineering College, Visakhapatnam, India.

B.P.Poorna satyanarayana is Associate Professor at Chaitanya Engineering College, Visakhapatnam, India