International Journal of Recent Trends in Engineering & Research
ISSN(ONLINE) : 2455-1457
IMPACT FACTOR : 3.344

Intruder Proof and Secure Cryptography

Arti
Extension lecturer,Deapartment of Computer Science,Government College for Girls,Sector-14,Gurgaon,Haryana,India

Abstract-Cryptography is a method of storing and transmitting data in a form that only those it is intended for can read and process. It is a science of protecting information by encoding it into an unreadable format. Although the ultimate goal of cryptography, and the mechanisms that make it up, is to hide information from unauthorized individuals, most algorithms can be broken and the information can be revealed if the attacker has enough time, desire, and resources. This paper presents the way of protecting the information from:

- Intruders: those who capture the packet and alter the information.
- Cryptanalysts: those who decrypt cipher text into plain text without key.

Keywords-Encryption, Cryptanalyst, Decryption, Intruder, Public key, Private key.

I. INTRODUCTION

From securing sensitive military information to securing personal messages, you often would be confronted with the need of masking information to protect it . And the method that provides security to messages is Cryptography. Cryptography comes from the Greek word “secret writing”. It is the process of combining some input data with a user-specified password to generate an encrypted output. The input is called the plain text and the output is called the cipher text. The encryption is done in such a way that no one can recover the original plain text without the encryption password in a reasonable amount of time. The algorithms that combine the keys and plain text to produce the cipher text are called ciphers . While cryptography is the science of securing data, cryptanalysis is the science of analyzing and breaking secure communication. The goals of cryptography are :

1. *Access Control* is the ability to limit and control the access to the system and application.
2. *Confidentiality* requires that a cryptanalyst will not be able to determine plain text from intercepted cipher text. There are numerous approaches to providing confidentiality, ranging from physical protection to mathematical algorithms which provide data unintelligible.
3. *Data integrity* is a service which addresses the unauthorized alteration of data. To assure data integrity, one must have the ability to detect data manipulation by unauthorized parties. Data manipulation includes such things as insertion, deletion, and substitution.
4. *Authentication* is to assure the recipient that the message is from the source that it claims to be from . Two parties entering into a communication should identify each other. Information delivered over a channel should be authenticated. Cryptography is usually subdivided into two major classes: *entity authentication* and *data origin authentication*.
5. *Data Integrity* addresses the unauthorized modification of data. One must have the ability to detect data manipulation by unauthorized parties such as insertion, deletion and substitution to ensure data integrity.
6. *Non-repudiation* is a service which prevents an entity from denying previous commitments or actions. When disputes arise due to an entity denying that certain actions were taken, a means to resolve the situation is necessary.

7. *Availability* requires that computer system possessions be available to authorized parties when needed.

Cryptography plays a very vital role in keeping the message safe as the data is in transit across insecure networks like Internet so that it cannot be read by anyone except the intended recipient. It ensures that the message being sent at one end remains confidential and should be received only by the intended receiver at the other end.

Cryptography referred almost exclusively to *encryption* that converts original message, called plain text, into non readable format, called cipher text and sends the message over an insecure channel. The reverse process of transforming cipher text messages back to plain text is called decryption.

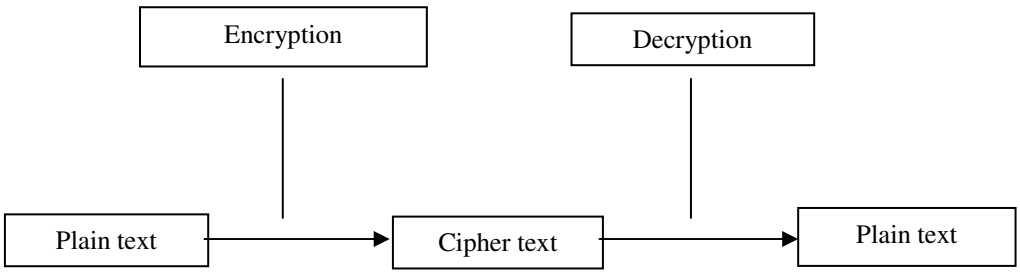

Fig. 1 Encryption and Decryption

A *cipher* (or *cypher*) is a pair of algorithms that create the encryption and the reversing decryption. The algorithm and a key control the operation of a cipher. This is a secret parameter known only to the communicants. Cryptography is a method that allows information to be sent in a secure form in such a way that only the intended receiver is able to retrieve this information. Presently continuous researches on the new cryptographic algorithms are going on. However, it is very difficult to find out the specific algorithm as they must consider many factors like security, features of algorithm, time complexity and space complexity. To ensure the security level of a cryptographic algorithm many effects have been made. Among of them avalanche, bit ratio, non-homogeneity, frequency distribution, time complexity are frequently used. The avalanche effect means a small change in plain text should create a significant change in cipher text. The bit ratio effect changes the bit values from same position between plain text and cipher text. The non-homogeneity test is a technique to test non-homogeneity of the source and encrypted file.

As the Internet and other forms of electronic communication become more prevalent, electronic security is becoming increasingly important. Cryptography is used to protect e-mail messages, credit card information, and corporate data. One of the most popular cryptography systems used on the Internet is Pretty Good Privacy because it's effective and free.

Cryptographic systems can be broadly classified as:

1. Symmetric-key systems:
   An encryption system in which the sender and receiver of a message share a single, common key that is used to encrypt and decrypt the message. Symmetric-key systems are simpler and faster, but their main drawback is that the two parties must somehow exchange the key in a secure way. Symmetric-key cryptography is sometimes called secret-key cryptography. The most popular symmetric-key system is the Data Encryption Standard (DES).
2. Public-key systems:
   A cryptographic system that uses two keys -- a public key known to everyone and a private or secret key known only to the recipient of the message. An important element to the public key system is that the public and private keys are related in such a way that only the public key can

be used to encrypt messages and only the corresponding private key can be used to decrypt them. Moreover, it is virtually impossible to deduce the private key if you know the public key. Public-key systems, such as Pretty Good Privacy (PGP), are becoming popular for transmitting information via the Internet. They are extremely secure and relatively simple to use. The only difficulty with public-key systems is that you need to know the recipient's public key to encrypt a message for him or her. Public key cryptography was invented in 1976 by Whitfield Diffie and Martin Hellman. For this reason, it is sometime called Diffie-Hellman encryption. It is also called asymmetric encryption because it uses two keys instead of one key (symmetric encryption)

II. EXISTING ALGORITHM FOR CRYPTOGRAPHY

2.1 Encryption algorithm

Data that can be read and understood without any special measures is called plaintext or cleartext. The method of disguising plaintext in such a way as to hide its substance is called encryption. Encrypting plaintext results in unreadable gibberish called ciphertext. Encryption is used to ensure that information is hidden from anyone for whom it is not intended, even those who can see the encrypted data. Algorithm for encrypting the plain text can be given as:

Step 1. open file fr in read mode
Step 2. read string plaintext
Step 3. set plaintext = plaintext.toUpperCase
Step 4. read shiftKey
Step 5. shiftKey = shiftKey % 26
Step 6. cipherText = ""
Step 7. set i=0
Repeat until i<plaintext.length()
Increment i by 1
Step 8. asciiValue = (int) plaintext.charAt(i)
Step 9. (asciiValue < 65 or asciiValue > 90){
cipherText += plaintext.charAt(i)
continue
Step 10. basicValue = asciiValue - 65
Step 11. newAsciiValue = 65 + ((basicValue + shiftKey) % 26)
Step 12. cipherText += (char) newAsciiValue
Step 13. print cipherText

2.2 Decryption algorithm

The process of reverting cipher text back into plaintext is decryption. The algorithm for decrypting the cipher text back into the plain text is given as:

Step 1. Open file fr in read mode
Step 2. Read string in ciphertext
Step 3. Convert ciphertext into upper case and store in ciphertext
Step 4. read as shiftKey
Step 5. shiftKey = shiftKey % 26;
Step 6. String plainText = "";

Step 7. i=0
Repeat until (i<ciphertext.length())
Increment i by 1
set asciiValue = (int) ciphertext.charAt(i);
Step 8. if (asciiValue < 65 or asciiValue > 90)
plainText += ciphertext.charAt(i);
continue;
Step 9. Set basicValue = asciiValue - 65;
Step 10.set newAsciiValue = -1000;
Step 11. if (basicValue - shiftKey < 0)
newAsciiValue = 90 - (shiftKey - basicValue) + 1
otherwise
Step 12. newAsciiValue = 65 + (basicValue - shiftKey)
Step 13. CONCATINATE plaintext with newAsciiValue
Step 14. PRINT plaintext

III. PROPOSED MODEL AND ITS IMPLEMENTATION

The information or plain text encrypted as cipher text is vulnerable to be accessed by intruders or cryptanalysts. The model presents the way of protecting the plain text from such unauthorized access.

3.1 Detection of Intruder's Activity

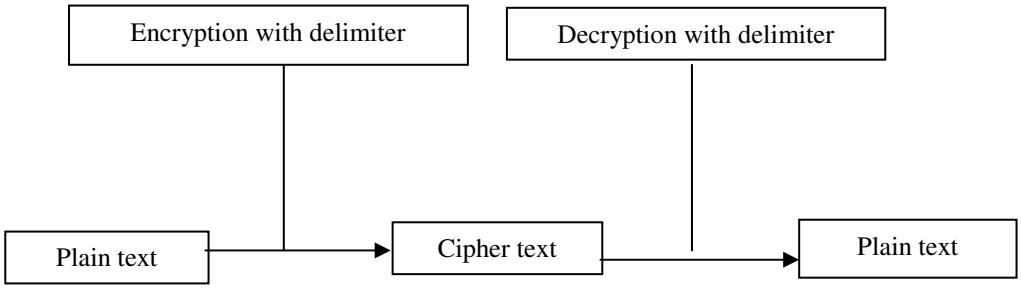

Fig 2. Encryption and decryption with delimiter

Some time intruder make changes in cipher text as a result it becomes difficult to know. Information becomes meaningless. A delimiter can be set with plain text during encryption so that at the time of decryption user could know whether cipher text was altered or not. If it is altered, delimiter will not be visible at receiving end and packet will be retransmitted

3.1.1 Algorithm for encryption

Step 1. open file fr in read mode
Step 2. read string plaintext
Step 3. set plaintext = plaintext.toUpperCase
Step 4. read shiftKey
Step 5. shiftKey = shiftKey % 26
Step 6. cipherText = ""
Step 7. set i=0
Repeat until i<plaintext.length()
Increment i by 1

Step 8. asciiValue = (int) plaintext.charAt(i)
Step 9. (asciiValue < 65 or asciiValue > 90){
cipherText += plaintext.charAt(i)
continue
Step 10. basicValue = asciiValue - 65
Step 11. newAsciiValue = 65 + ((basicValue + shiftKey) % 26)
Step 12. cipherText += (char) newAsciiValue
Step 13. add delimeter to ciphertext
Step 13. print cipherText

3.1.2 Algorithm for decryption

Step 1. Open file fr in read mode
Step 2. Read string in ciphertext
Check the delimiter in ciphertext
If delimiter not found then
Print "INVALID DATA"
AND EXIT
OTHERWISE
Step 3. Convert ciphertext into upper case and store in ciphertext
Step 4. read as shiftKey
Step 5. shiftKey = shiftKey % 26;
Step 6. String plainText = "";
Step 7. i=0
Repeat until (i<ciphertext.length())
Increment i by 1
set asciiValue = (int) ciphertext.charAt(i);
Step 8. if (asciiValue < 65 or asciiValue > 90)
plainText += ciphertext.charAt(i);
continue;
Step 9. Set basicValue = asciiValue - 65;
Step 10.set newAsciiValue = -1000;
Step 11. if (basicValue - shiftKey < 0)
newAsciiValue = 90 - (shiftKey - basicValue) + 1
otherwise
Step 12. newAsciiValue = 65 + (basicValue - shiftKey)
Step 13. CONCATINATE plaintext with newAsciiValue
Step 14. PRINT plainText

3.2 Privileged Cryptography with IP Filtering

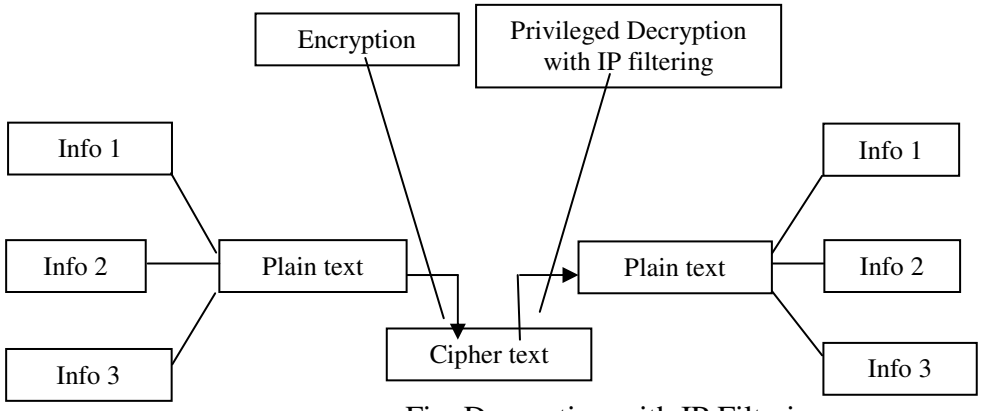

Fig. Decryption with IP Filtering

To protect information from cryptanalyst IP Filter would be attached in decryption module. Sometime cryptanalyst decrypt information without key, this problem can be solved using IP filtering. If there is decryption request from different IP then packets will be deleted.

3.2.1 On sender side

1. Read info1, info2, info3.
2. And merge then to create plain text.
3. Encrypt the text

3.2.2 On receiver side

1. Check whether IP address is valid or not
2. If IP address in invalid then skip otherwise
3. Decrypt the data and convert it into plain text .
4. Check the user level
5. If user level is A then display info1.
6. If user level is B then display info1 , info2
7. If user level is C then display info1, info2, info3
8. If user level is D then display info1, info2, info3, info4.

IV. RESULT

It will improve the security of the text while sending it from sender to receiver. A delimiter can be set during encryption and decryption to know whether the data has been altered by an intruder or not. We can also protect our information from cryptanalyst, who can decrypt information without using key, by attaching IP filter during decryption. Hence our data becomes more secure when transmitted from sender to receiver.

REFERENCES

[1] V. Boone, *A Brief History of Cryptology*, Naval Institute Press, Annapolis, Maryland, 2005.
[2] Preeti Gulia and Prof. Dr. R.S. Chillar, *Cryptographic Techniques: An Overview*, National Conference on Emerging Trends in Mobile Technologies and Security, March 29, 2011, pp. 278-282.
[3] T.H. Barr, *Invitation to Cryptology*, Prentice Hall, Upper Saddle River, New Jersey, 2002

[4] G.B. Agnew, *Random Sources for Cryptographic Systems*, Advances in Cryptology- Eurocrypt'87 Proceedings, Springer-Verlag, 1988, pp. 77-81.
[5] T. El-Mageed, N. Hamdy, F. Amer and Y. Kerisha, *Cipher System and Cryptanalysis Techniques: An overview of the basic principles*, The Egyptian Computer Journal, ISSR, Cario Univ, vol(28), no. 1, 2000.
[6] W. Stallings, *Cryptography and Network Security: Principles and Practice*, Prentice-Hall, USA, Second Edition, 1999.
[7] Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.
[8] Monika Agarwal and Pradeep Mishra, *A Comparative Survey on Symmetric Key Encryption Techniques*, International Journal on Computer Science and Engineering, vol. 4 no. 05 May 2012.
[9] Vishwas Gupta, Gajendra Singh and Ravindra Gupta, *Advance Cryptography Algorithm for improving data security*, International Journal of Advanced Research in Computer Science and Software Engineering, vol. 2, issue 1, January 2012.
[10] Uttam Kr. Mondal, Satyendra Nath Mandal, J. Pal Choudhary, J.K. Mandal, *Frame Based Symmetric Key Cryptography*, International Journal of Advanced Networking and Applications, vol. 02, issue 04, 2011, pp. 762-769

