A PATTERN RECOGNITION APPROACH TO BLOCK CIPHER IDENTIFICATION

A THESIS

submitted by

SREENIVASULU NAGIREDDY

for the award of the degree

of

MASTER OF SCIENCE

(by Research)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

October 2008

Dedicated to my parents

THESIS CERTIFICATE

This is to certify that the thesis entitled **A Pattern Recognition Approach to Block Cipher Identification** submitted by **Sreenivasulu Nagireddy** to the Indian Institute of Technology Madras, for the award of the degree of Master of Science (By Research) is a bonafide record of research work carried out by him under our supervision and guidance. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. C. Chandra Sekhar

Dr. Hema A. Murthy

Chennai-600036

Date:

ACKNOWLEDGEMENTS

I wish to express my gratitude to everyone who contributed in making this work a reality. There are persons not mentioned here but, deserve my token of appreciation, first of all, I thank them all. Let me also thank the Indian Institute of Technology Madras for giving me an opportunity to pursue my post graduation. I am grateful to Computer science department for providing the facilities for the same. This institute has given me great exposure. There has been a drastic change in my vision of life. I enjoyed every moment of my IIT life.

I would like to thank my guides, Prof. Hema A. Murthy and Dr. C Chandra Sekhar whose expertise, understanding, and patience, helped me through the course. I also thank Dr. Shri Kant (DRDO) for clearing my concepts and for his valuable suggetions in solving complex problems. I am grateful to the other members of my general test committee, Dr. N S Narayanaswamy and Dr. C S Ramalingam for the assistance they provided at all levels of my research. I also thank other faculty members of Computer science department especially, Dr. Debdeep Mukhopadhyay for taking time out from his busy schedule for reviewing my work and giving precious feedbacks. My sincere gratitude to the head of the computer science department, Prof. Timothy A Gonsalves for the facilities and support for the completion of my course. I hereby, thank every faculty member of this department who taught me many values in life apart from the regular curriculum especially Dr. Shankar Balachandran who is very friendly and source inspiration for my research. I thank my friend Swapna who was also working in similar research topic and have given valuable suggestions in my research as well

GENERAL TEST COMMITTEE

1 **Chairperson:** Dr. P Sreenivas Kumar

2 **Guides:** Prof. Hema A. Murthy and Dr. C. Chandra Sekhar

3 **Members:**

- Dr. N.S. Narayanaswamy
- Dr. C.S. Ramalingam

personal life. My lab-mates have been extraordinarily helpful, in both my academic and personal life. Kasirajan, Deivapalan, Manohar, Nikhil, Samuel, Dinil and lot many at DON lab have made their remarkable contributions for the fruitful results of my research. My friends at IIT, Manikumar, Sheetal, Noor, Mahesh, Anil, Ramakrishna, Prashant, Bhanu, Sarma, Laskhmi Narayana, Abhilash, Sid and KP without whom my IIT life is unimaginable.

I express my salutation to god almighty for giving me such understanding and supporting parents and family.

ABSTRACT

Keywords: *Cryptanalysis; Identification of encryption method; Pattern recognition techniques; Histogram method; Support Vector Machines; Gaussian Mixture Models; Data expansion methods;Block ciphers; AES; DES; BlowFish; TDES; RC5*

With the advent of high speed network technologies, large amount of sensitive data is being transfered over the Internet and other public channels. Faster and larger storage devices have become cheaper. The concern for protection of information is increasing at an alarming rate. Block ciphers are being widely used for secure transmission over the network and also for storing information securely in fixed hard drives and removable drives. New block ciphers are being designed as the existing block ciphers are prone to attack and also to make the encryption process faster than the existing ciphers. Users of block ciphers must be convinced that the block ciphers that they are using are resilient to different kinds of attacks. Hence, a mechanism is required to assess the strength of the block ciphers. Identification attack or distinguishing attack is a cryptanalytic attack that serves as a primary assessment of strength of ciphers. This attack can be applied prior to either the key space reduction or in decryption without the key.

Most of the research in cryptanalysis starts with the assumptions that the encryption method and mode of operation are already known. Also researchers work on reduced versions of encryption algorithm instead of full algorithm. But in real world it is necessary to know the encryption method and mode of operation before doing the

CURRICULUM VITAE

1. **Name:** Sreenivasulu Nagireddy
2. **Date of Birth:** 25^{th} May, 1982
3. **Educational Qualifications:**
   - 2008 - Master of Science (M.S)
   - 2003 - Bachelor of Engineering (B.E)
4. **Permanent address:**

   Srikolanu, Sangam,

   Nellore - 524308,

   Andhra Pradesh.

   Email-id: sreeni.nr@gmail.com

cryptanalysis of finding key or plaintext. In this work we try to find out encryption method and mode of operation. Further we use the complete encryption algorithms instead of simplified versions.

Identification of the encryption method of block ciphers is a kind of cryptanalysis attack. It is considered to be the primary attack that assesses the security of the block ciphers. Two sub-problems of this task namely, (a) detecting the encryption mode and (b) locating the starting position of a block in a captured ciphertext are addressed in this work.

Identification of the encryption method from block ciphers is considered as a pattern classification problem. We use pattern recognition techniques such as the histogram matching method, support vector machines and Gaussian mixture models for detecting the patterns in block ciphers. We develop several attacking techniques under three attacking models (a) ciphertext-only attack, (b) known-plaintext attack and (c) side channel attack.

Various techniques have been proposed and implemented as part of this research. The results indicate that cipher texts in the ECB mode are prone to attack where as the CBC mode cipher texts are robust. The AES emerges as a strong block cipher among the widely used block ciphers such as AES, DES, TDES, Blowfish and RC5.

TABLE OF CONTENTS

[57] Serge Vaudenay, "On the weak keys of blowfish," in *Proceedings of the Third International Workshop on Fast Software Encryption*, London, UK, 1996, pp. 27–32, Springer-Verlag.

[58] "Rc5," http://en.wikipedia.org/wiki/RC5.

[59] Alex Biryukov and Eyal Kushilevitz, "Improved cryptanalysis of rc5," *Lecture Notes in Computer Science*, vol. 1403, pp. 85–100, 1998.

[60] "Project rc5 by distributed.net," http://www.distributed.net/rc5/.

[47] D. W. Davies and S. Murphy, "Pairs and triplets of des s-boxes," *Journal of . Cryptology*, vol. 8, pp. 1–25, 1995.

[48] Eli Biham and Alex Biryukov, "An improvement of davies' attack on des," *Journal of Cryptology: the journal of the International Association for Cryptologic Research*, vol. 10, no. 3, pp. 195–205, 1997.

[49] Ralph C. Merkle and Martin E. Hellman, "On the security of multiple encryption," *Commun. ACM*, vol. 24, no. 7, pp. 465–467, 1981.

[50] C. van Oorschot and Michael J. Wiener, "A known-plaintext attack on two-key triple encryption," in *EUROCRYPT '90: Proceedings of the workshop on the theory and application of cryptographic techniques on Advances in cryptology*, New York, NY, USA, 1991, pp. 318–325, Springer-Verlag New York, Inc.

[51] Stefan Lucks, "Attacking triple encryption," in *FSE '98: Proceedings of the 5th International Workshop on Fast Software Encryption*, London, UK, 1998, pp. 239–253, Springer-Verlag.

[52] Eli Biham, "How to forge des-encrypted messages in 2^{28} steps," .

[53] Daniel J. Bernstein, "Cache-timing attacks on AES," 2004.

[54] D. J. Bernstein., "Cache-timing attacks on aes. technical report," http://cr.yp.to/antiforgery/cachetiming-20050414.pdf, 2005.

[55] D. Osvik, A. Shamir, and E. Tromer, "Cache attacks and countermeasures: the case of aes," 2005.

[56] Bruce Schneier, "Description of a new variable-length key, 64-bit block cipher (blowfish)," in *Fast Software Encryption, Cambridge Security Workshop*, London, UK, 1994, pp. 191–204, Springer-Verlag.

models for text categorization," in *Workshop on Language Modeling and Information Retrieval*, 2001.

[38] R. Cilibrasi and P.M.B. Vitanyi, "Clustering by compression," *Information Theory, IEEE Transactions on*, vol. 51, no. 4, pp. 1523–1545, April 2005.

[39] A. Viterbi, "Convolutional codes and their performance in communication systems," *Communications, IEEE Transactions on [legacy, pre - 1988]*, vol. 19, no. 5, pp. 751–772, Oct 1971.

[40] "Statistical distributions of english text," http://www.data-compression.com/ english.html.

[41] Mitsuru Matsui, "Linear cryptanalysis method for des cipher," in *EUROCRYPT '93: Workshop on the theory and application of cryptographic techniques on Advances in cryptology*, Secaucus, NJ, USA, 1994, pp. 386–397, Springer-Verlag New York, Inc.

[42] "Openssl project," http://www.openssl.org/.

[43] Gilles Garon and Richard Outerbridge, "Des watch: an examination of the sufficiency of the data encryption standard for financial institution information security in the 1990s," *SIGSAC Rev.*, vol. 9, no. 4, pp. 29–45, 1991.

[44] Eli Biham and Adi Shamir, *Differential cryptanalysis of the data encryption standard*, Springer-Verlag, London, UK, 1993.

[45] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosystems," *Journal of Cryptology*, vol. 4, no. 1, pp. 3–72, 1991.

[46] E. Biham and A. Shamir., "Differential cryptanalysis of the full 16-round des," in *Advances in Cryptology - Crypto '92*. 1993, pp. 487–496, Springer-Verlag.

[29] Ronan Collobert and Samy Bengio, "Svmtorch: support vector machines for large-scale regression problems," *J. Mach. Learn. Res.*, vol. 1, pp. 143–160, 2001.

[30] Satish Kumar, *Neural Networks - A Class Room Approach*, Tata McGraw-Hill, New Delhi, 2004.

[31] Klaus-Robert Müller, Alex J. Smola, Gunnar Rätsch, Bernhard Schölkopf, Jens Kohlmorgen, and Vladimir Vapnik, "Predicting time series with support vector machines," in *ICANN '97: Proceedings of the 7th International Conference on Artificial Neural Networks*, London, UK, 1997, pp. 999–1004, Springer-Verlag.

[32] Andrew Moore, "Gaussian mixture models," http:// www.autonlab.org/ tutorials/gmm.html.

[33] T.K. Moon, "The expectation-maximization algorithm," *Signal Processing Magazine, IEEE*, vol. 13, no. 6, pp. 47–60, Nov 1996.

[34] L. R. Bahl, M. Padmanabhan, D. Nahamoo, and P. S. Gopalakrishnan, "Discriminative training of gaussian mixture models for large vocabulary speech recognition systems," in *ICASSP '96: Proceedings of the Acoustics, Speech, and Signal Processing, 1996. on Conference Proceedings., 1996 IEEE International Conference*, Washington, DC, USA, 1996, pp. 613–616, IEEE Computer Society.

[35] D.A. Reynolds and R.C. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, Jan 1995.

[36] Haim Permuter, Joseph Francos, and Ian Jermyn, "A study of gaussian mixture models of color and texture features for image classification and segmentation," *Pattern Recogn.*, vol. 39, no. 4, pp. 695–706, 2006.

[37] William J. Teahan and David J. Harper, "Using compression-based language

LIST OF TABLES

[18] Karine Gandolfi, Christophe Mourtel, and Francis Olivier, "Electromagnetic analysis: Concrete results," in *CHES '01: Proceedings of the Third International Workshop on Cryptographic Hardware and Embedded Systems*, London, UK, 2001, pp. 251–261, Springer-Verlag.

[19] Alan G. Konheim, *Cryptography: A Primer*, New York: John Wiley & Sons Inc, 1981.

[20] "Reuters corpus," http://about.reuters.com /researchandstandards /corpus.

[21] William M. Fisher, George R. Doddington, , and Kathleen M. Goudie-Marshall, "The darpa speech recognition research database: Specifications and status," in *Proceedings of DARPA Workshop on Speech Recognition*, 1986, pp. 93–99.

[22] "Iitm generic object image library," http://vplab.cs.iitm.ernet.in/.

[23] National Bureau of Standards, "Data encryption standard," *Federal Information Processing Standards Publication (FIPS PUB) No. 46*, 1977.

[24] "Advanced encryption standard fips-197," *National Institute of Standards and Technology*, 2001.

[25] Bruce Schneier, "The blowfish encryption algorithm," http:// www.schneier.com/ blowfish.html.

[26] William C. Barker, "Recommendation for triple data encryption algorithm," *NIST Special Publications 800-67 Version1*, 2004.

[27] Ronald L. Rivest, "The rc5 encryption algorithm," in *Fast Software Encryption*, 1995, pp. 86–96.

[28] Abraham Sinkov, *Elementary cryptanalysis: A Mathematical Approach*, The Mathematical Association of America, 1966.

/library/crypto.html.

[8] Bruce Schneier, *Applied cryptography (2nd ed.): protocols, algorithms, and source code in C*, John Wiley & Sons, Inc., New York, NY, USA, 1995.

[9] R. A Rueppel, "Stream ciphers in contemporary cryptography," *The Science of Information Integrity, Editor - G. Simmons*, pp. 65–134, 1991.

[10] Randall K. Nichols and Panos Lekkas, *Wireless Security: Models, Threats, and Solutions*, McGraw-Hill Professional, 2002.

[11] National Bureau of Standards, "Des modes of operation," *Federal Information Processing Standards Publication (FIPS PUB) 81*, 1980.

[12] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.

[13] Bruce Schneier, "A self-study course in block-cipher cryptanalysis," *Cryptologia*, vol. XXIV, no. 1, pp. 18–33, 2000.

[14] Douglas R. Stinson, *Cryptography: Theory and Practice*, CRC Press, Inc., Boca Raton, FL, USA, 1995.

[15] Paul C. Kocher, "Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems," *Lecture Notes in Computer Science*, vol. 1109, pp. 104–113, 1996.

[16] Paul Kocher, Joshua Jaffe, and Benjamin Jun, "Differential power analysis," *Lecture Notes in Computer Science*, vol. 1666, pp. 388–397, 1999.

[17] Suresh Chari, Charanjit Jutla, Josyula R. Rao, and Pankaj Rohatgi, "A cautionary note regarding evaluation of AES candidates on smart-cards," in *Second Advanced Encryption Standard (AES) Candidate Conference*, Rome, Italy, 1999, http://csrc.nist.gov/encryption/aes/round1/conf2/aes2conf.htm.

BIBLIOGRAPHY

[1] Matt Blaze, "A cryptographic file system for unix," in *CCS '93: Proceedings of the 1st ACM conference on Computer and communications security*, New York, NY, USA, 1993, pp. 9–16, ACM.

[2] Yuliang Zheng and Yongge Wang, "Efficient and provably secure ciphers for storage device block level encryption," in *StorageSS '05: Proceedings of the 2005 ACM workshop on Storage security and survivability*, New York, NY, USA, 2005, pp. 103–107, ACM.

[3] Xun Yi, Chik How Tan, Chee Kheong Slew, and M. Rahman Syed, "Fast encryption for multimedia," *IEEE Transactions on Consumer Electronics*, vol. 47, no. 1, pp. 101–107, Feb 2001.

[4] A.D. Dileep and C.C. Sekhar, "Identification of block ciphers using support vector machines," *International Joint Conference on Neural Networks, 2006. IJCNN '06.*, pp. 2696–2701, 16-21 July 2006.

[5] Eli Biham, "Cryptanalysis of multiple modes of operation," in *ASIACRYPT '94: Proceedings of the 4th International Conference on the Theory and Applications of Cryptology*, London, UK, 1995, pp. 278–292, Springer-Verlag.

[6] Keliher L., Meijer H., and Tavares S., "Provable security of substitution-permutation encryption networks against linear cryptanalysis," *Canadian Conference on Electrical and Computer Engineering*, vol. 1, pp. 37–42 vol.1, 2000.

[7] Gary C. Kessler, "An overview of cryptography," http://www.garykessler.net

A.5.1 RC5 Operational Overview

Unlike many other block ciphers, RC5 has a variable block size (32, 64 or 128 bits), key size (0 to 2040 bits) and number of rounds (0 to 255). The RC5 is word oriented, a block consists of two words. RC5 with parameters represented as RC5-$w/r/b$. w, r, b stands for word size, number of rounds and number of bytes in key respectively. The nominal choice of parameters proposed are a block size of 64 bits (word length of 32 bits), a 128-bit key and 12 rounds [27] referred as RC5 32/12/16. The main feature of RC5 is the intensive use of data-dependent rotations. RC5 also consists of a number of modular additions and XORs. The general structure of the algorithm is a Feistel-like network. The encryption and decryption routines can be specified in a few lines of code. The key schedule, however, is more complex. The tantalizing simplicity of the algorithm together with the novelty of the data-dependent rotations has made RC5 an attractive object of study for cryptanalysts [58].

A.5.2 Cryptanalysis of RC5

RC5 32/12/16 is susceptible to a differential attack using 2^{44} chosen plaintexts [59]. 18 to 20 rounds are suggested as sufficient protection. distributed.net has successfully brute forced 56 and 64-bit key RC5 messages [60]. This is the the largest successful publicly-known brute force attack as of 2008.

LIST OF FIGURES

XORed to produce the final 32-bit output.

Since Blowfish is a Feistel network, it can be inverted simply by XORing P_{17} and P_{18} to the ciphertext block, then using the P-entries in reverse order.

Blowfish's key schedule starts by initializing the P-array and S-boxes with values derived from the hexadecimal digits of π, which contain no obvious pattern. The secret key is then XORed with the P-entries in order (cycling the key if necessary). A 64-bit all-zero block is then encrypted with the algorithm as it stands. The resultant ciphertext replaces P_1 and P_2. The ciphertext is then encrypted again with the new subkeys, and P_3 and P_4 are replaced by the new ciphertext. This continues, replacing the entire P-array and all the S-box entries.

A.4.2 Cryptanalysis of Blowfish

Though there is no effective cryptanalysis on the full-round version of Blowfish known publicly as of 2008, there is a known plaintext attack found by Serge Vaudenay requiring 2^{8r+1} known plaintexts to break, where r is the number of rounds [57]. Moreover, he also found a class of weak keys that can be detected and broken by the same attack with only 2^{4r+1} known plaintexts. This attack cannot be used against the regular Blowfish; as it assumes the knowledge of the key-dependent S-boxes.

A.5 RC5

RC5 is a very simple and fast block cipher suitable for both hardware and software implementations. It was designed by Ronald Rivest in 1994 [27] for RSA Data Security. RC stands for "Rivest Cipher", or alternatively, "Ron's Code".

A.4 Blowfish

Blowfish is a Feistel cipher designed in 1993 by Bruce Schneier and included in a large number of encryption products. Blowfish provides a good encryption rate in software and no effective cryptanalysis of it has been found to date. Schneier designed Blowfish cipher as a drop-in replacement for DES or IDEA. Blowfish is unpatented and Bruce Schneier has stated that "Everyone is welcome to download Blowfish and use it in their application". Therefore algorithm is in the public domain, and can be freely used by anyone [25].

It is significantly faster than DES when implemented on 32-bit microprocessors with large data caches, such as the Pentium and the PowerPC [56]. Notable features of the design include key-dependent S-boxes and a highly complex key schedule. Key scheduling can be a considerable overhead. Hence Blowfish is suitable for applications where the key does not change often, like a communications link or an automatic file encryptor.

A.4.1 Blowfish Operational Overview

Blowfish has a 64-bit block size and the key can be any length up to 448 bits [56]. It is a 16-round Feistel cipher and uses large key-dependent S-boxes.

The algorithm of Blowfish keeps two subkey arrays: the 18-entry P-array and four 256-entry S-boxes. The S-boxes accept 8-bit input and produce 32-bit output. One entry of the P-array is used every round, and after the final round, each half of the data block is XORed with one of the two remaining unused P-entries. Blowfish algorithm has an F-function. The function splits the 32-bit input into four eight-bit quarters, and uses the quarters as input to the S-boxes. The outputs are added modulo 2^{32} and

CHAPTER 1

Introduction

1.1 Overview

Block ciphers are not only used for secure transmission over network but also for storing information securely in fixed hard drives and removable drives [1] [2]. There are many block ciphers available today. New block ciphers are being designed (a) as the existing block ciphers are prone to attack and (b) to make the encryption process faster [3]. Users of these block ciphers must be convinced that the block ciphers that they are using are resilient to all kinds of attacks. Hence, a mechanism is required to assess the strength of block ciphers. Identification attack or distinguishing attack is a cryptanalytic attack that serves as a primary assessment of ciphers. This attack can be applied prior to either key space reduction or in decryption without the key as shown in Figure 1.1.

Identification of encryption method of block ciphers is a kind of cryptanalysis attack. It is considered as a primary attack that assesses the security of the block ciphers. This work address the following two subproblems: (a) detecting the encryption mode and (b) locating the starting position of a block of the captured ciphertext.

Two major stages in the encryption method identification systems are as follows: modeling stage of the encryption methods, where the distinct features of an encryption method are determined and stored in models, and the classification stage in which an

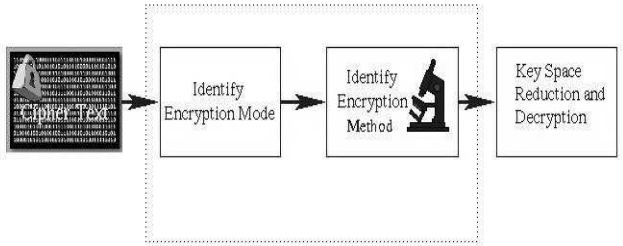

Fig. 1.1: Preprocessing before key-space reduction and decryption in cryptanalysis

input ciphertext is compared to the models of the encryption method. The encryption method model that is the most similar to the input ciphertext represents the encryption method of the ciphertext.

In this thesis identification of the encryption method for block ciphers is considered as a pattern classification problem. Pattern classification techniques that use the histogram matching method, support vector machines (SVM) and Gaussian mixture models (GMM) are considered for this task. Techniques are developed under following attacking models (a) ciphertext-only attack, (b) known-plaintext attack and (c) side-channel attack.

Ciphertexts encrypted using the block cipher encryption methods usually contain uniform distributions of symbols. Finding the distinguishable patterns in ciphertexts is the main focus of the thesis. Various methods have been proposed in this work for block cipher identification in ECB and CBC modes of encryption. Detailed explanation on the modes of encryption is given in Chapter 2.

Identification of an encryption method from a ciphertext is considered as a classification task [4]. Figure 1.2 shows the basic mechanism of this cryptanalytic attack.

We developed the following methods for identification of the encryption method

A.3.1 AES Operational Overview

AES has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits, whereas Rijndael can be specified with key and block sizes in any multiple of 32 bits, with a minimum of 128 bits and a maximum of 256 bits. Due to the fixed block size of 128 bits, AES operates on a 44 array of bytes, termed the state. Most AES calculations are done in a special finite field [24].

A.3.2 Cryptanalysis of AES

So far, the only successful attacks against AES implementations have been side channel attacks. Side channel attacks do not attack the underlying cipher and so have nothing to do with its security as described here, but attack implementations of the cipher on systems which inadvertently leak data. There are several such known attacks on certain implementations of AES [53].

In April 2005, D.J. Bernstein announced a cache timing attack that he used to break a custom server that used OpenSSL's AES encryption [54]. The custom server was designed to give out as much timing information as possible, and the attack required over 200 million chosen plaintexts. In October 2005, Dag Arne Osvik, Adi Shamir and Eran Tromer presented a paper demonstrating several cache timing attacks against AES [55]. One attack was able to obtain an entire AES key after only 800 operations triggering encryptions, in a total of 65 milliseconds. This attack requires the attacker to be able to run programs on the same system that is performing AES.

A.2.2 Cryptanalysis of TDES

The best attack known on 3-key TDES requires around 2^{32} known plaintexts, 2^{113} steps, 2^{90} single DES encryptions, and 2^{88} memory [51]. This is not currently practical. If the attacker seeks to discover any one of many cryptographic keys, there is a memory-efficient attack which will discover one of 2^{28} keys, given a handful of chosen plaintexts per key and around 2^{84} encryption operations [52].

TDES is slowly disappearing from use, largely replaced by its natural successor, the Advanced Encryption Standard (AES). One large-scale exception is within the electronic payments industry, which still uses 2TDES extensively and continues to develop and promulgate standards based upon it (e.g. EMV(Europay, MasterCard and Visa)). This guarantees that TDES will remain an active cryptographic standard well into the future. TDES suffers from slow performance in software; on modern processors, AES tends to be around six times faster.

A.3 Advanced Encryption Standard (AES)

The Advanced Encryption Standard (AES), also known as Rijndael, is a block cipher adopted as an encryption standard by the U.S. government. It has been analyzed extensively and is now used worldwide, as was the case with its predecessor, the DES. The Advanced Encryption Standard, became the official successor to DES in December 2001 [24].

Unlike DES, its predecessor, Rijndael is a substitution-permutation network, not a Feistel network. AES is fast in both software and hardware, is relatively easy to implement, and requires little memory. As a new encryption standard, it is currently being deployed on a large scale.

Fig. 1.2: Identification of encryption method from a ciphertext

of the ciphertexts encrypted with block ciphers; histogram method, block prediction method, data expansion techniques and side channel techniques. All the techniques other than the side-channel techniques follow the basic principle of finding unique patterns in the ciphertexts. Histogram method works based on the assumption that histograms of ciphertexts for different encryption methods vary significantly and this phenomenon can be captured. Block prediction method is designed especially for analyzing the ciphertext under CBC(cipher-block chaining) mode of operation. The CBC mode attermpts to remove the patterns that would occur otherwise in the ECB (electronic codebook) mode of operation. We show how the CBC and ECB mode ciphertexts can be differentiated by means entropy of ciphertexts. In the ciphertexts encrypted under CBC mode of operation, every block is dependent upon all the previous blocks [5]. So we exploit the strength of relationship between a block and its previous blocks in this method. Data expansion methods are helpful in exposing the patterns in the ciphertexts by expanding them. We also address the issue of missing starting bytes of a ciphertext using histogram method, as it would be very difficult to analyze such ciphertexts directly. The block ciphers are too nonlinear in nature, to identify. We need some other information such as the side-channel information to identify the encryption method.

1.2 Motivation

A block cipher using a block length of k bits is designed such that, it can take any of the 2^k possible input texts and produce one of the 2^k possible output texts [6]. A secret key is involved in the encryption process which decides the mapping function of the input space to the output space. Since the output text for a block cipher has 2^k possible values, it is very difficult to identify the encryption method of any ciphertext. However, in a practical scenario the input text provided by a user to the encryption method does not occupy the entire input space, as it contains only meaningful text (readable alphanumeric characters). In other words, the input space is much smaller than the possible input space. Hence the output space is also limited and varies based on the encryption method and the secret key involved. So, it is assumed that the encryption method can be identified by analyzing the ciphertext.

The requirement for protection of information in the fields like communication and storage is increasing rapidly. In some applications, such as pay-TV, medical imaging systems, military image/database communications, and confidential video conferences, highly secure and reliable storage and transmission of multimedia data is needed. Web pages and multimedia content have a low entropy and are prone to identification attack. To encrypt large data such as multimedia data, parallel encryption is used to minimize the number of accesses to the storage device and to maximize CPU utilization. Among all the modes of encryption, only the ECB mode of encryption supports parallelism, whereas other modes such as CBC, CFB (Cipher Feedback) and OFB(Output Feedback) work serially. In our work, we show that the ECB mode of encryption is vulnerable to attacks in many block cipher encryption methods. So, a proper encryption method has to be chosen for securing information and for fast encryption and decryption.

A.2 Triple DES

Triple DES overcomes many of the problems faced by DES. Triple DES or TDES is a block cipher formed from the DES cipher by using it three times. It is three times slower than regular DES but can be billions of times more secure if used properly. When it was found that a 56-bit key of DES is not enough to guard against brute force attacks, TDES was chosen as a simple way to enlarge the key space without a need to switch to a new algorithm.

A.2.1 TDES Operational Overview

The simplest variant of TDES operates as follows: $DES(k3; DES(k2; DES(k1; M)))$, where M is the message block to be encrypted and $k1$, $k2$, and $k3$ are DES keys. This variant is commonly known as EEE because all three DES operations are encryptions. In order to simplify interoperability between DES and TDES the middle step is usually replaced with decryption (EDE mode): $DES(k3; DES^{-1}(k2; DES(k1; M)))$ and so a single DES encryption with key k can be represented as $TDES - EDE$ with $k1 = k2 = k3 = k$. The choice of decryption for the middle step does not affect the security of the algorithm.

In general TDES with three different keys (3-key TDES) has a key length of 168 bits: three 56-bit DES keys (with parity bits 3-key TDES has the total storage length of 192 bits). But due to the meet-in-the-middle attack the effective security it provides is only 112 bits. A variant, called 2-key TDES, uses $k1 = k3$, thus reducing the key size to 112 bits and the storage length to 128 bits. However, this mode is susceptible to certain chosen-plaintext or known-plaintext attacks [49] [50].

substitutes [23].

A.1.2 Cryptanalysis of DES

Although more information has been published on the cryptanalysis of DES than any other block cipher, the most practical attack to date is still a brute force approach. There are three attacks known that can break the full sixteen rounds of DES with less complexity than a brute-force search: differential cryptanalysis (DC), linear cryptanalysis (LC), and Davies' attack. However, these attacks are theoretical and are not feasible to mount in practice.

Differential cryptanalysis was introduced in the late 1980s by Eli Biham and Adi Shamir [44], [45]. To break the full 16 rounds, differential cryptanalysis requires 2^{47} chosen plaintexts [46]. Although it is a theoretical breakthrough, this attack is not practical because of both the large data requirements and the difficulty of mounting a chosen plaintext attack. DES was designed to be resistant to DC.

Linear cryptanalysis was discovered by Mitsuru Matsui, and needs 2^{43} known plaintexts [41]. The first experimental cryptanalysis of DES was successfully achieved in an attack requiring 50 days on 12 HP 9735 workstations. However, this attack is still impractical.

While linear and differential cryptanalysis are general techniques and can be applied to a number of schemes, Davies' attack [47] is a specialized technique for DES, first suggested by Donald Davies in the eighties, and improved by Biham and Biryukov [48]. The most powerful form of the attack requires 2^{50} known plaintexts, with a computational complexity of 2^{50}, and 51% success rate.

Most of the research in cryptanalysis starts with the assumption that the encryption method and mode of operation are already known. Also, researchers work on the simplified versions of encryption algorithms. In real world, it is necessary to know the encryption method and mode of operation before doing the cryptanalysis of finding the key or plaintext. In this work, we try to identify the encryption method and mode of operation and we deal with the complete encryption algorithms instead of their simplified versions.

1.3 Major contributions

The following are the major contributions of this thesis:

1. Developing a mechanism to assess the strength of the block ciphers
2. Designing techniques using pattern recognition methods such as the histogram matching method, support vector machines and Gaussian mixture models for capturing patterns in the ciphertexts and therefore identifying the encryption method of the ciphertexts
3. Analyzing different modes of encryption in block ciphers
4. Comparing the widely used block ciphers such as DES, AES, Blowfish, TDES and RC5 using the techniques for identification attack.

1.4 Organization

The rest of the thesis is organized as follows.Chapter 2 provides an overview of cryptography and cryptanalysis. It describes various types of cryptography and introduces the block ciphers and its modes of encryption. It also presents cryptanalysis and different

attacking models of cryptanalysis.

Chapter 3 describes the proposed approaches for solving the identification problem using the ciphertext-only attack model. The approaches include the histogram based matching method, Hamming distance based method, the data expansion method and the block prediction method. It also gives solution to the sub-problems such as finding the mode of encryption and locating the starting position of a block in a given ciphertext when some of the initial bytes are missed.

Chapter 4 discusses various approaches for identification of encryption method of block ciphers using the known-plaintext attack and side-channel attack models. It also explains how support vector regression is used for predicting plaintext from ciphertext. Two methods under the side-channel attack model such as timing attack and architectural attack are described here.

Chapter 5 summarizes the research work carried out, highlights contributions of the work and gives pointers to further research in this field.

APPENDIX A

Block Ciphers

A.1 Data Encryption Standard(DES)

In1974, the National Bureau of Standards(NBS) (now National Institute of Standards and Technology(NIST)) called for possible encryption algorithms to use in unclassified government communications. IBM submitted their Lucifer design. Which after a period of redesign, it became the Data Encryption Standard (DES). It was officially accepted as standard in 1977 by NBS [23]. DES is probably the most widely used secret key system, particularly in securing financial data. One of the largest users of the DES is the banking industry, particularly with Electronic funds transfer (EFT), and EFTPOS(Electronic Fund Transfer at Point Of Sale). Automated Teller Machines (ATMs) typically use DES [43].

A.1.1 Operational Overview of DES

DES uses a 56-bit key. The 56-bit key is divided into eight 7-bit blocks and an 8th odd parity bit is added to each block (i.e., a 0 or 1 is added to the block so that there are an odd number of 1 bits in each 8-bit block). These 8 parity bits are used for rudimentary error detection. Hence a DES key is actually 64 bits in length for computational purposes (although it only has 56 bits worth of randomness). DES acts on a 64-bit blocks of the plaintext, invoking 16 rounds of permutations, swaps, and

5.2 Scope for future work

- Extending this work to streams ciphers and public key cryptography.

CHAPTER 2

Overview of Cryptography

2.1 Introduction

This chapter gives an overview of cryptography and cryptanalysis. In particular, it discusses the different types of cryptographic algorithms namely, stream ciphers and block ciphers. It also gives a brief overview of different types of cryptanalytic techniques. Cryptography deals with securing information while cryptanalysis verifies the promises of cryptographic algorithms.

2.2 Cryptography

Cryptography deals with developing methods for taking a legible, readable plaintext, and transforming it into an unreadable form. The main purpose of cryptography is secure transmission and storing of sensitive information. The process of transforming a plaintext to ciphertext is called encryption. The reverse process of converting a ciphertext to the plaintext is called decryption.

Protecting information is an important problem. In [7], Gary Kessler identifies four security requirements for cryptography:

- Authentication: The process of identifying an individual.
- Privacy/Confidentiality: Making sure that the information will be used or read

by the intended users only.

- Integrity: Integrity, in terms of data and network security, is the assurance that the information can only be accessed or modified by those authorized to do so. Information cannot be altered in storage, or in transit between the sender and the intended receiver, without the alteration being detected.
- Non-repudiation: A mechanism to ensure that a transferred message has been sent and received by the parties claiming to have sent and received the message. Non-repudiation is a way to guarantee that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message.

The terms cryptographic algorithms and ciphers are used interchangeably in the rest of the thesis. The message to be encrypted is called a plaintext and the output of the encryption process is called a ciphertext.

2.3 Properties of a good modern cipher

The following are the desired properties of a good modern cipher:

- Large size key to resist brute-force search
- Fast encryption process
- Published and reviewed algorithm: Ensuring that security depends entirely on the secrecy of key but not depend on the algorithm secrecy. (It implies that key or plaintext can not be obtained from ciphertext just by going through the algorithm of encryption method)
- Resistance to chosen-plaintext attack: Attacker cannot determine key even if

mode of ciphers.

- We provide a mechanism to assess the strength of block ciphers
- We provide a mechanism that can be used as the preprocessing step before the key space reduction or decryption of a ciphertext without knowing the secret information.
- Histogram method can be used to identify the encryption method of ciphertexts in the ECB mode under the cipher-text only attack model.
- It is proved in our work that the ECB mode of encryption does not provide serious message confidentiality, and hence it is not recommended for use in secure transmission of sensitive data.
- We conclude that the pattern recognition techniques can not be used to detect patterns in CBC mode ciphertexts of DES, AES, Blowfish, TDES and RC5.
- Multiple key attack is proposed as a mechanism to assess the strength of the encryption method in case of small plaintext being encrypted with several keys.
- We compared the widely used encryption methods DES, AES, Blowfish, TDES and RC5 using the results obtained from different techniques for identification attack. We found that the AES encryption method is stronger block cipher compared to the others.

The techniques suggested in this work add to the pool of cryptanalytic attacks on the block ciphers. When the new ciphers are introduced, the designers have to take such attacks into consideration in order to design safe primitives and should aim at resistance against these attacks.

CHAPTER 5

Conclusions

5.1 Conclusions

In this work, we explored various techniques for identification of the encryption method of the ciphertexts encrypted using the block ciphers. We described our analysis on why the ECB mode ciphertexts can be identified better than the CBC mode ciphertexts. Our results can be summarized as follows. The histogram method is extremely powerful in identifying the encryption method of the ciphertexts in ECB mode. The block prediction method is developed to exploit the ciphers in the CBC mode, but results show that the ciphertexts in CBC mode are indeed very robust. However considering only the first block of the ciphertexts in the CBC mode with the constant initialization vector and large number training samples it is possible to identify the encryption method of a given ciphertext using the histogram method. Locating starting position of a block in the captured ciphertext is also dealt in this work successfully for the ECB mode ciphers. We also showed how the ECB and CBC mode ciphertexts can be distinguished by means of the entropy of the ciphertext. Even in the known-plaintext attack model, it is not possible to identify CBC mode ciphertexts using the pattern recognition methods. We also devised the side-channel attacks for identification of the encryption method such as the timing attack and architectural attack which gives good performance in identifying the encryption method regardless of the encryption

attacker can choose the plaintext for encryption. It implies resistance to known-plaintext and known-ciphertext attacks

- Resistance to distinguishing attack: It should not produce patterns in the ciphertext output.

In this work we verify the property, "resistance to distinguishing attack" of cipher systems.

2.4 Types of cryptography

There are several ways of classifying the cryptographic algorithms. Based on the number of keys used in encryption and decryption there two types of cryptography:

- Symmetric cryptography or private-key cryptography
- Asymmetric cryptography or public-key cryptography

In symmetric cryptography or private-key cryptography, the same key is used for both encryption and decryption. This means that the encryption key must be shared between the two parties before any messages can be decrypted. Symmetric cryptography can be used to transmit information over an insecure public channel. It has also other uses, such as secure storage on insecure media and strong mutual authentication.

In asymmetric cryptography or public-key cryptography two different keys, public key and private Key are used. The private key is kept secret while the public key is distributed. Anyone can encrypt a message but only the one who knows the corresponding private key can decrypt it. Though the keys are mathematically related, it is practically not possible to retrieve the private key from the public key. The public key is used for encryption and one can decrypt the ciphertext only with the private key.

The main disadvantage of the symmetric cryptography is the key distribution. Key must be exchanged with the receivers every time the key is changed for encryption. So, the key is not changed often in symmetric cryptography. The main disadvantage of asymmetric cryptography is that these algorithms are very slow. Symmetric cryptography is widely used now a days due to its fast encryption process and portability [8]. In this thesis symmetric cryptographic algorithms are analyzed.

2.4.1 Types of symmetric cryptography

Symmetric cryptography is further subdivided into the following two types based on the amount of information it can encrypt or decrypt at a time:

- Stream ciphers.
- Block ciphers.

Stream ciphers: Stream ciphers are symmetric ciphers which encrypt one bit or one byte at a time.

Block ciphers: A block-cipher divides the plaintext into blocks of bits and uses a specially constructed function which mixes a block of the plaintext with the secret key to produce a block of the ciphertext. A stream cipher can be treated as a block cipher with a really small block size. Rueppel [9] summarizes the distinction between the block ciphers and stream ciphers as follows; "Block ciphers operate with a fixed transformation on large blocks of plaintext data; stream ciphers operate with a time-varying transformation on individual plaintext digits." Block ciphers are the most widely used ciphers [10]. This thesis focuses on the block ciphers and their analysis.

4.3 Summary

In this chapter we have described techniques for identifying the encryption method in two different attacking models known-plaintext attack and side channel attack. We conclude that the CBC mode ciphertexts are robust enough to known-plaintext attacks also. Side channel attacks can easily determine the encryption method as it deals with the physical implementation of the encryption methods.

Table 4.3: Time taken for different sizes of data by DES, AES, Blowfish and TDES. Time is shown in microseconds, SD- standard deviation

Method	Data Size	Avg Time Taken	SD	Min Time	Max Time
DES	1 MB	40683	7.07	37088	44896
AES	1 MB	21379	10.09	16592	25372
Blowfish	1 MB	25324	8.61	21472	30256
TDES	1 MB	91543	4.93	86864	95648
DES	2 MB	72926	6.13	68320	80032
AES	2 MB	33775	8.32	29280	40992
Blowfish	2 MB	40846	5.53	37088	43920
TDES	2 MB	173740	2.78	169824	176656
DES	4 MB	137898	3.31	134688	141521
AES	4 MB	60325	4.81	55632	65392
Blowfish	4 MB	76520	5.05	72224	81008
TDES	4 MB	342465	2.04	338672	345505

Table 4.4: Number of times the bitwise instructions appear in footprints of algorithms for DES, AES, Blowfish and TDES

Instruction	Number of Times Called			
	DES	AES	Blowfish	TDES
XOR	410	219	151	410
AND	132	73	72	138
SHR	64	85	70	66
ADD	1	27	98	1

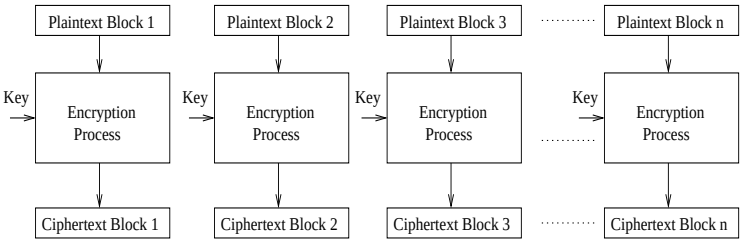

Fig. 2.1: ECB mode encryption

2.5 Block cipher modes of operation

Block ciphers operate on blocks of fixed length data, often 64 or 128 bits. Since the plaintext to be encrypted can be of any length, several schemes or encryption modes are used for encrypting messages larger than the block size [11]. Some of these encryption modes are briefly described below.

2.5.1 Electronic Codebook (ECB) mode

Electronic codebook mode is the simplest mode of encryption. In this mode of encryption, a plaintext is divided into fixed size blocks and each block is encrypted separately. Each identical block of a plaintext gives an identical block of a ciphertext. In this mode ciphertexts may contain some patterns, which makes the cryptanalytic attacks easier. Although this is the most common mode of encryption, it is susceptible to different kinds of attack. In this thesis, we show the weakness of ECB mode ciphers towards the identification attack. The main advantage of ECB mode of encryption is that it enables parallelism in encrypting the plaintext blocks and decrypting the ciphertext blocks. This yields high performance. Figure 2.1 shows how ECB mode encryption is performed and Figure 2.2 depicts the decryption process.

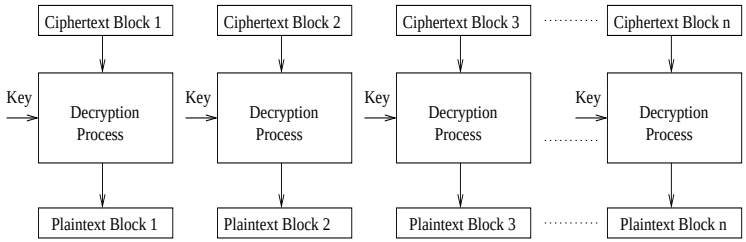

Fig. 2.2: ECB mode decryption

2.5.2 Cipher Block Chaining (CBC) Mode

In this mode, a plaintext is divided into fixed size blocks and each plaintext block is XORed with the previous ciphertext block before being encrypted. This means that the ciphertext for a particular block depends not only on the block itself, but also on all the previous plaintext blocks. An initialization vector (IV) is used to XOR the first plaintext block before encrypting it. To make each message unique, a different IV should be used each time [12]. The IV does not need to be secret. It is only intended as an extra random factor to disrupt the dictionary attacks and similar. It is usual to prepend the IV to the beginning of the ciphertext, so that the decryptor knows what to use. Knowledge of the IV is not a lot of help to anyone attacking the encryption scheme. The CBC mode ciphers are most nonlinear in nature and do not reveal any patterns in the ciphertext. In this thesis we show that the CBC mode ciphers are more secure compared to the ECB mode ciphers. The disadvantage of the CBC mode encryption is that it will not allow parallelization during encryption. So it may not be suitable for large data encryption when time taken is critical. Figure 2.3 shows how the CBC mode encryption is performed and Figure 2.4 depicts the decryption process.

4.2.1 Timing attack

Timing attack is a side channel attack in which the attacker attempts to compromise a cryptosystem by analyzing the time taken to execute cryptographic algorithms. The attack exploits the fact that every operation in a computer takes some time to execute. Here we applied the timing attack on encryption algorithms implemented in *OpenSSL 0.9.7a* [42] for identifying the encryption algorithm. We show that the time taken for encrypting a particular amount of data is different for different encryption methods. Every method is tested 1000 times, each time with different key for accuracy. Time taken for various sizes of data is shown in Table 4.3. It can be inferred from the table that the encryption method can be identified clearly if the data size is greater than 4 MB from the timing information.

4.2.2 Architectural attack

In this attack, the cryptanalyst takes advantage of having access to the cache/ PROM /ROM/ storage media where the actual executable code for the encryption method is stored. If we look at the lower level implementation details of the encryption algorithms, they perform bitwise operations such as XOR, OR and AND. We count the number of times these instructions are used in the code. Based on the nature of the algorithm, each algorithm uses these instructions different number of times. Table 4.4 provides the number of times, some of the bit-wise instructions are used in the executable code. By examining the number of times the bitwise instructions are called in the executable code, we can identify the cryptosystem.

Side channel attacks perform well in identifying the block ciphers because the implementation of block ciphers varies largely.

termine the functionality even when the data is transformed into higher dimensions. We conclude that the pattern recognition techniques are not suitable for capturing the mapping functionality in block ciphers.

4.2 Side channel attacks

All the earlier techniques that use ciphertext and plaintext data have not been so useful in identifying the CBC mode ciphers. We require some other information which will help in identifying the encryption method. In this section, we present the side channel attacks which work irrespective of the mode of operation.

A side channel attack is an attack based on the information gained from the physical implementation of a cryptosystem, rather than the theoretical weakness in the algorithms. This information is denoted as "Side Channel Information". Side channel information is the information that can be obtained from the encryption/decryption device that is neither plaintext to be encrypted nor ciphertext resulting from the encryption process. Side channel attacks are gaining importance as the cryptographic algorithms are usually strong against mathematical attacks. Their practical implementation, both in software and in hardware, opens the door to side-channel attacks.

Side channel information that is considered in our study is the time taken for the encryption process and the executable code of the encryption method. These side-channel attacks require considerable technical knowledge of the internal operation of the system on which the encryption method is implemented. Side channel attacks are quite successful but they are less practical than the ciphertext-only attack and the known-plaintext-attack.

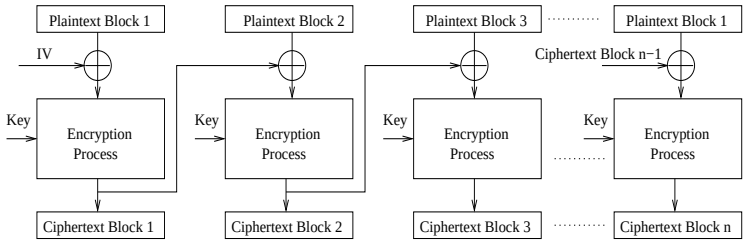

Fig. 2.3: CBC mode encryption

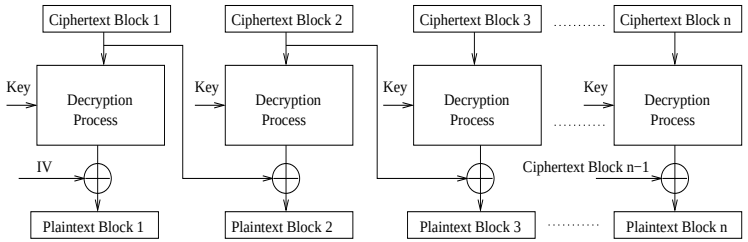

Fig. 2.4: CBC mode decryption

2.6 Cryptanalysis

Cryptanalysis is the art and science of analyzing and breaking ciphers. "Breaking ciphers does not necessarily mean finding a practical way for an eavesdropper to recover the plaintext from just the ciphertext. Breaking a cipher simply means finding a weakness in the cipher that can be exploited with a complexity less than brute-force [13]." Many of the trusted encryption algorithms have been made available to public scrutiny to find and correct any weakness in them. Exploiting the weakness(if any) of ciphers and showing a way to build new stronger ciphers is the main purpose of cryptanalysis. Cryptanalysis verifies the promises of the cryptography. The various techniques in cryptanalysis attempting to compromise the cryptosystems are referred to as attacks. well known cryptanalytic attacks of widely used block ciphers such as DES, AES, Blowfish, 3DES and RC5 are explained in Appendix A.

2.6.1 Types of Cryptanalysis Attacks

Based on the information that an attacker or cryptanalyst can obtain, the cryptanalysis attacks can be classified as follows:

2.6.1.1 Ciphertext-only attack

A ciphertext-only attack is a cryptanalysis attacking scheme in which the cryptanalyst has access only to a set of ciphertexts. It is quite difficult to attack any cipher in this scheme. Ciphertext-only attacking model is used to attack the classical ciphers. Most of the modern ciphers are not vulnerable to this type of attack. Since the attacker has access to only the ciphertext, entropy plays a major role in this type of attack. In this scheme, the data is relatively easy to obtain in many scenarios, but a successful

Table 4.1: Performance (in %) of the known plaintext attack with Support vector regression method for identification of the encryption method for CBC mode. Parameters: st - parameter std in rbf kernel , c - trade-off between training error and margin

st	c	DES	AES	Blowfish	TDES	RC5	Avg
80.00	90.00	20.00	30.00	25.00	15.00	15.00	21.00
50.00	90.00	22.50	30.00	25.00	10.00	22.50	20.00
30.00	90.00	22.50	25.00	12.50	20.00	25.00	21.00
50.00	10.00	30.00	20.00	25.00	15.00	20.00	22.00
50.00	500.00	27.50	22.50	17.50	15.00	25.00	21.50

Table 4.2: Performance (in %) of the known plaintext attack based on binary properties with SVM classifier for identification of the encryption method. Parameters: st - parameter std in rbf kernel , c - trade-off between training error and margin

st	c	DES	AES	Blowfish	TDES	RC5	Avg
10.00	100.00	17.50	27.50	22.50	10.00	22.50	20.00

this technique the feature set consists of parity, number of ones and checksum of both plaintext block and ciphertext blocks. The feature set consists of 6 elements with 3 features extracted from plaintext and 3 from ciphertext. The SVM classifier is used in this technique for classifying the ciphertexts. Table 4.2 shows classification performance for this method.

The use of plaintexts has not helped in identifying the encryption method using SVMs. In block ciphers, the encryption process or mapping function from plaintext to ciphertext is complex as the plaintext goes through many rounds of substitutions, permutations and key mixing. Pattern recognition techniques like SVM cannot de-

Ciphertext only attack the attacker requires only ciphertext. Where as in case of KPA, the attacker requires both ciphertext and corresponding plaintext. Chosen plaintext is even less practical because attacker requires access to the encryption device to encrypt his/her chosen plaintext.

4.1.1 Mapping from plaintext to ciphertext

Having the training data of both plaintext and ciphertext, we map every block of plaintext to its corresponding ciphertext block using support vector regression(SVR) technique. During the training phase, the mapping models are built for the encryption methods. Classification in this method is performed as follows. Given a ciphertext and plaintext pair for identification, we determine the mapping model that suits best for that pair using SVR. Corresponding encryption method is chosen as the encryption method of the given plaintext and ciphertext pair. Feature selection is an important criteria that influences classification in pattern recognition techniques, In this method feature vector is plaintext and its corresponding ciphertext block represented in binary form. Instead of 1's and 0's the bits are represented as series of 1's and -1's for better classification performance with SVR. Table 4.1 shows the classification results.

4.1.2 Classification using parity and binary properties of data block

Motivation for this method is to capture the relationship between the characteristics of the plaintext and ciphertext such as checksum, number of bits set and parity. We conjecture that the relationship between plaintext and ciphertext properties such as parity, number of bits set and checksum vary for different encryption methods. In

ciphertext-only attack is generally difficult, and requires a very large ciphertext sample. It is almost impossible to break the ciphers which produce high entropy ciphertexts in this scheme. Here the objective of the cryptanalyst is to find the key or determining the plaintext from a ciphertext [14].

2.6.1.2 Known-plaintext attack

A known-plaintext attack is one in which the cryptanalyst has access to a sample of ciphertext and the corresponding plaintext as well. Here the objective of the cryptanalyst is to find the key. A ciphertext only attack is easier to apply in the real world than known-plaintext attack. And also it is less practical in the real world to apply, as it requires both the plaintext and the ciphertext for analysis [14].

2.6.1.3 Chosen-plaintext attack

A chosen-plaintext attack is an attack where the cryptanalyst is able to define his own plaintext, feed it into the cipher, and analyze the resulting ciphertext. Applying a chosen-plaintext attack requires the cryptanalyst be able to send data of his choice into the device which is doing the encryption, and it requires the cryptanalyst to be able to view the output from the device. Because of these requirements, a chosen-plaintext attack is impossible to attempt in some cases [14].

2.6.1.4 Chosen-ciphertext attack

In this case, the cryptanalyst chooses a ciphertext and causes it to be decrypted with an unknown key. It is similar to the chosen-plaintext attack. Both the chosen-ciphertext attack and the chosen-plaintext attack are less practical [14].

2.6.1.5 Side channel attack

This attacking model does not deal with the ciphertext or plaintext. A side channel attack is any attack based on information gained from the physical implementation of a cryptosystem, rather than theoretical weakness in the algorithm. Here, the cryptosystem is compromised with the attacker. The side channel information can be the time taken to encrypt or decrypt, power consumption or sound produced during computation by computer workstations. First proposed in 1996 [15], side-channel attacks have since been used to extract the cryptographic key material of symmetric and public key encryption algorithms running on microprocessors, DSPs, FPGAs, ASICs and high performance CPUs using side-channel information such as variations in power consumption [16] [17], time delay [15] or electromagnetic radiation [18].

2.7 Summary

The overview of the cryptography and cryptanalysis is briefly explained in this chapter. Different types ciphers and the modes of operation are presented. Cryptanalysis with different of types of attacking models is discussed briefly.

CHAPTER 4

Known-plaintext and Side-channel Attacks

Ciphertext-only attacks are not helpful in identifying the encryption method of ciphertexts in the CBC mode. Hence It requires some other information along with the ciphertexts. This chapter provides description of the known-plaintext and side-channel attacks for the identification of the encryption method of ciphertexts. Known-plaintext attack requires plaintext of the corresponding ciphertext. Side channel attacks requires information of physical implementation of the encryption algorithms.

4.1 Known plaintext attack with support vector machines

The known-plaintext attack (KPA) is an attack model for cryptanalysis where the attacker has samples of both the plaintext and its encrypted version (ciphertext) and is at the liberty to make use of them to reveal further secret information about key. It is more practical than the chosen-plaintext but less practical than the ciphertext-only attack. Most of the attacks in the literature are of this type. The most notable attack of this kind is the linear cryptanalysis [41]. Practicality is defined based on how easy attacker can obtain required information and can attack the system. In case of

3.11 Summary

In this chapter we have seen various cipher-text only attack techniques for identifying the encryption method of the block ciphers. The histogram method can be used for identifying the encryption method well in ECB mode ciphers. But it fails to identify the CBC mode ciphers. This method also has been used to locate the starting position of block in cropped ciphertexts. As the histogram method failed in case of CBC mode ciphers, we developed other techniques such as the block prediction method, GMM based method, Hamming distance based methods and data expansion methods. But all these methods appear to give more or less the same poor performance. Finally we analyzed and described the reasons behind the high classification rate for the ECB mode ciphers.

CHAPTER 3

Ciphertext-only Attack

3.1 Overview

In cryptography, a ciphertext-only attack or known ciphertext attack is an attacking model for cryptanalysis where the attacker is assumed to have access only to a set of ciphertexts. This attack is considered to be the most difficult and the most practical attack in cryptanalysis. Ciphertext-only attacks are the most useful attacks in cryptosystems, since they require only passive eavesdropping from the attacker. Such attacks are usually hard to find, since the assumptions on the knowledge of attacker are minimal [19].

In this attack model, a large number of ciphertexts are available to the attacker and their corresponding encryption method is also known apriori. We use statistical and pattern recognition methods such as histogram method, support vector machines (SVM), and Gaussian mixture model(GMM) method for identifying the encryption method of ciphertexts. Subproblems of identification of the encryption method such as determining the mode of encryption and finding the starting position of block are also dealt.

The proposed techniques are explained in the following sections. Histogram method is described in Section 2. Multiple and bitwise histogram techniques are explained in Sections 3, and Section 4 respectively. Block prediction method is discussed in Sec-

tion 5. Gaussian mixture model technique is described in Section 6. Section 7 talks about Hamming distance based classification techniques. Data compression and expansion techniques are explained in Section 8. An analysis of the strength of block ciphers in multiple key attack is explained in Section 9. Finally the results obtained are summarized.

3.1.1 Data sets used in the study

The plaintext data used in this study is extracted from Reuters Corpus [20]. Multimedia data is also encrypted and tested in some experiments. We used multimedia data from DARPA speech database (TIMIT) [21] and IITM Generic Object Image Library [22]. Each dataset is divided into 100 files for training and 40 files for testing. The file size is set to 512 bytes for the block prediction method and 10KB for the histogram based methods. The histogram based methods requires more training data because the training data should be sufficiently large to distinguish the frequency of occurrence of each symbol. Size and number samples of data decides the success rate of the attack. Due to the computational limitations we have considered 16bit histograms in case of histogram based methods. Analyzing histograms at higher granular level with more samples will definitely reveal more patterns.

The plaintexts are encrypted with several different keys and the results shown in this thesis are average of experiments of the different sets of ciphertext data. Keys used in generating the test data are the same as that used for training. The five encryption methods considered in this study are: DES(Data encryption standard) [23], AES(Advanced Encryption Standard) [24], Blowfish [25], TDES(Triple DES) [26] and RC5(RC stands for "Rivest Cipher") [27]. The detailed explanation of these encryption methods is given in Appendix A.

Table 3.19: Performance (in %) of the Histogram method for identification of encryption algorithm in ECB mode. Keeping some of the bits constants

Number of bits constant	Encryption Method					Average
in each character	DES	AES	Blowfish	TDES	RC5	
1	17.50	17.50	30.00	15.00	17.50	18.50
2	25.00	22.50	15.00	20.00	20.00	21.50
4	25.00	32.50	17.50	20.00	20.00	23.00

that mere two constant bits in plaintext is not resulting high classification rate in the ECB mode. English plaintext also contains many other patterns that synthetically generated plaintext does not possess. There are some words in English that occur very frequently and also many words in English have common prefixes and suffixes. Some characters such as 'e' , 't' and 's' occur more frequently than other characters. Likewise, ST, NG, TH, and QU are common pairs of letters that occur together (termed bigrams or digraphs) [40]. Due to these properties of English language, the plaintext has very low entropy and produces patterns in ciphertexts in the ECB mode for weak block ciphers. AES ciphertext in the ECB mode are comparatively secure (see Figures 3.1, 3.2, 3.3, 3.4 and 3.5). AES is a strong encryption method due to its well designed substitution-permutation network (SPN) based encryption process and also has larger key size and block size. But Encryption methods having smaller block length are susceptible to pattern in plaintext being reflected in ciphertext.

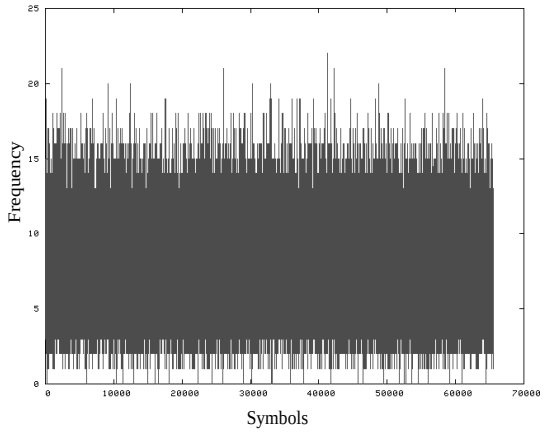

Fig. 3.24: Histogram of DES ciphertexts in ECB mode of synthetic data keeping 2 bit constant in each character.

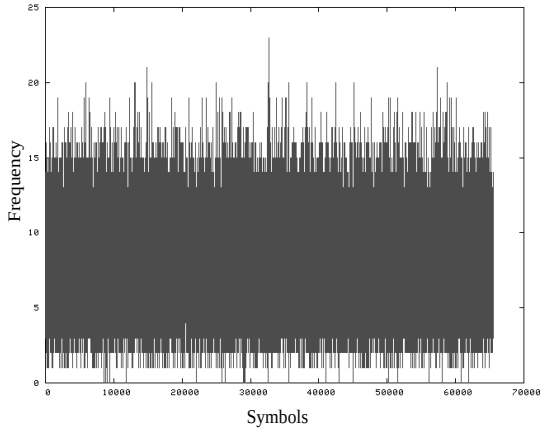

Fig. 3.25: Histogram of DES ciphertexts in ECB mode of synthetic data keeping 4 bit constant in each character..

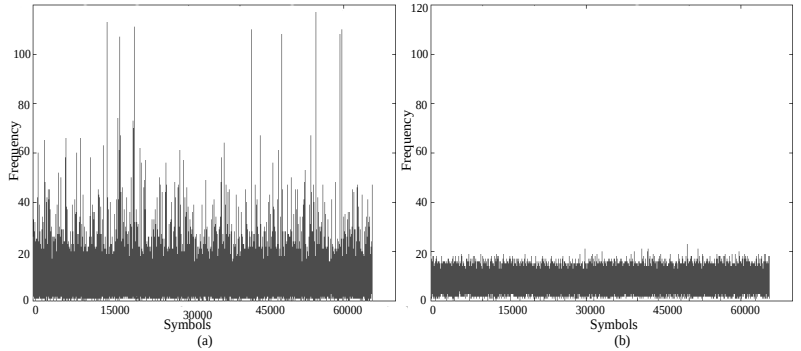

Fig. 3.1: Histograms for ciphertexts of DES in (a) ECB mode and (b) CBC mode

3.2 Histogram method

Histogram method is similar to the frequency analysis technique which was widely used in breaking the classical ciphers in the cipher-text-only attack model [28]. The main objective of this method is to capture the statistical properties of ciphertexts. A histogram shows variations in the frequency of occurrence of symbols, which can be used as the classification criterion. It is seen that patterns can be observed in the histograms of the ciphertexts of different encryption methods. An encryption method attempts to obtain a nearly uniform distribution of the symbols that can occur in the ciphertexts. Hence, the process of identifying the encryption method from a ciphertext is a difficult task. However, fine nuances in the histograms for the ciphertexts of different encryption methods can be captured.

Figures 3.1, 3.2, 3.3, 3.4 and 3.5 show the histograms for ciphertexts obtained using DES, AES, Blowfish, TDES and RC5 encryption methods in the ECB and CBC modes. The ECB mode is as secure as the underlying block cipher. Patterns are not revealed in the histograms of ciphertexts for the ECB mode of the AES encryption

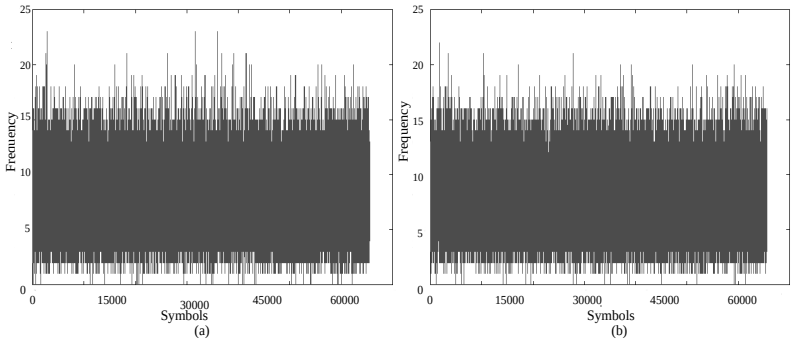

Fig. 3.2: Histograms for ciphertexts of AES in (a) ECB mode and (b) CBC mode

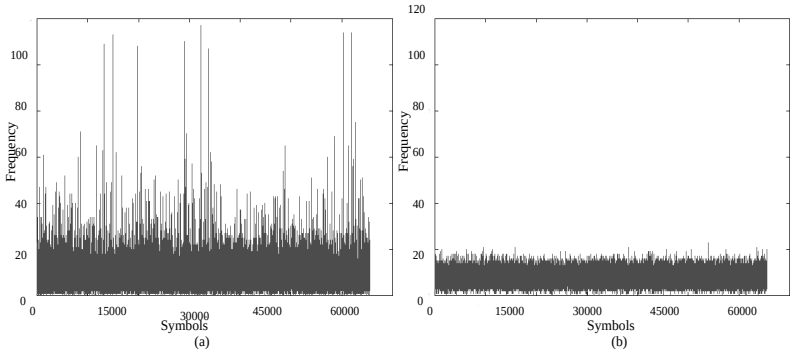

Fig. 3.3: Histograms for ciphertexts of Blowfish in (a) ECB mode and (b) CBC mode

mode ciphertexts compared to the CBC mode ciphertexts. It is mainly because of the high entropy of CBC mode ciphertexts. The main difference in input to the encryption operation between the ECB and CBC is that, in the ECB mode of encryption the plaintext is given as input directly, whereas in case of the CBC mode input text is XORed with previous ciphertext block. In general plaintext of English text data has low entropy and some of the bits in the plaintext are always constant. Consider the block size of 64 bit, it contains 8 characters of plaintext. The ASCII values of alphanumeric characters ranges from 48 to 126. There is one bit common in all these characters. That is 8^{th} bit and its value is 0. So the total block of 8 characters contains 8 constant bits.

In the ECB mode, encryption method encrypts data of k characters having $2k$ bits constants always. For the DES, the block size is 64bit and 16 bits are constant in every block of plaintext. Whereas in the CBC mode the chaining effect makes no bits constant. So we conjecture that these constant bits are perhaps causing patterns in the ciphertext.

To ensure that only these two constant bits and not any other factors are causing patterns in ciphertext the following experiment is done. We generated synthetic plaintext by keeping some of the bits constant and rest of the bits were randomly varied with uniform probability.

Figures 3.24 and 3.25 show the histograms of DES ciphertexts in ECB mode for the generated plaintext keeping two bits and four bits constant respectively. Histograms of ciphertexts for encryption methods such as AES, Blowfish, TripleDES and RC5 are also similar to the DES.

Table 3.19 gives the classification result for identification of the encryption method for ciphertext of synthetic data. The classification results from the Table 3.19 shows

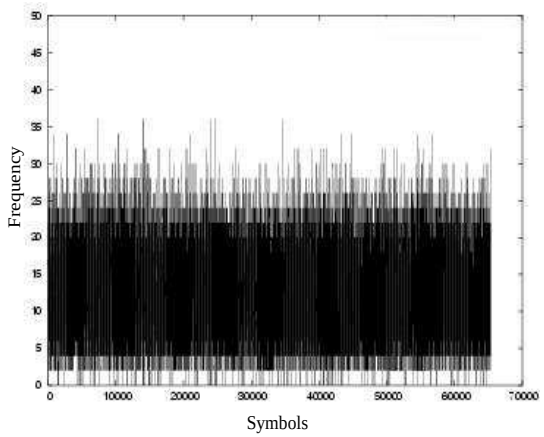

Fig. 3.22: Histogram of 16,00,00 bytes of TripleDES ciphertext resulting from encryption of 16 byte number with 1,00,000 keys.

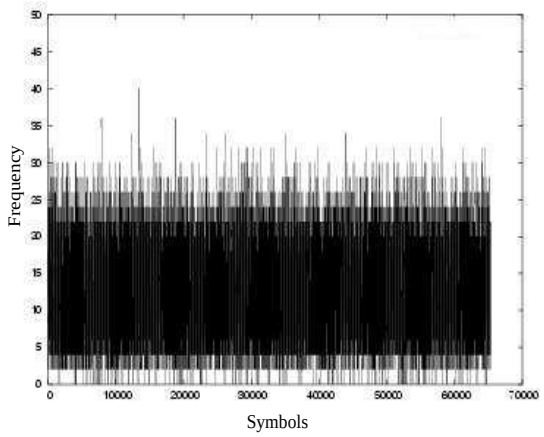

Fig. 3.23: Histogram of 16,00,00 bytes of RC5 ciphertext resulting from encryption of 16 byte number with 1,00,000 keys.

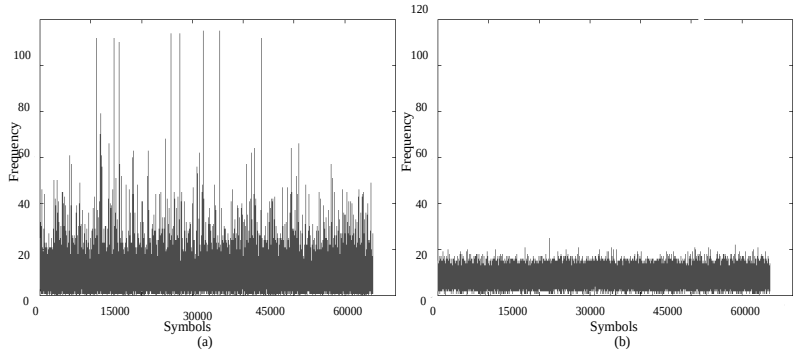

Fig. 3.4: Histograms for ciphertexts of TDES in (a) ECB mode and (b) CBC mode

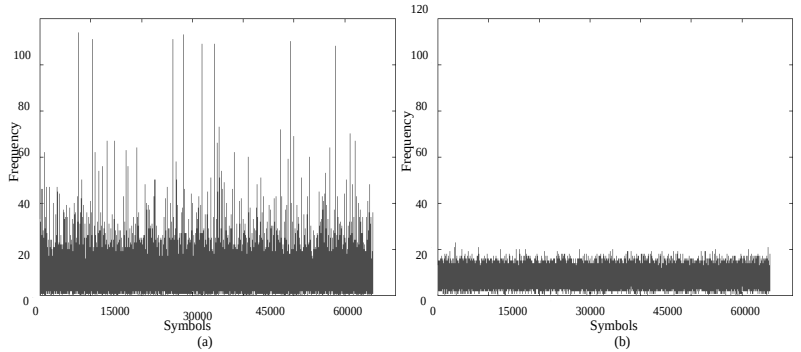

Fig. 3.5: Histograms for ciphertexts of RC5 in (a) ECB mode and (b) CBC mode

method while the other methods such as DES, Blowfish, TDES and RC5 reveal some patterns in their histograms of the ECB mode ciphertexts. The CBC mode ciphertexts of all the five methods do not reveal any patterns.

3.2.1 Histogram based classification

In this method, the probability of occurrence of each symbol in the ciphertext is considered as a feature vector. If the symbol size is l bits, the possible number of different symbols is 2^l. Histogram based method depends on the amount of training data available. It works quite well for the ECB mode of encryption compared to the CBC mode of encryption because, the variations in histograms of the ECB mode ciphertexts are high compared to the histograms of CBC mode ciphertexts. The block diagram of the histogram based classifier is shown in Figure 3.6. For each encryption method, the probability of occurrence of each symbol is computed from a large amount of training data. A test ciphertext is classified based on the nearest Euclidean distance of its histogram to the histograms of training data of different encryption methods.

Classification performance for the ECB and CBC modes of encryption are given in Tables 3.1 and 3.2 for different symbol sizes(l). Since the histograms for the ECB mode ciphers contain high variations, the ciphers in this mode are identified accurately. The histograms for the CBC mode ciphers are almost uniformly distributed and therefore it is not possible to distinguish one from the other. It is evident from the results that, the increase in the symbol size improves the performance in the case of ECB mode asthe frequency variation become clearly visible with an increase in the symbol size. Increasing the symbol size is not helpful in the case of CBC mode ciphers.

Classification performance for the training data that contains ciphertexts, generated using 5 different keys and test data that contains ciphertexts, generated using the

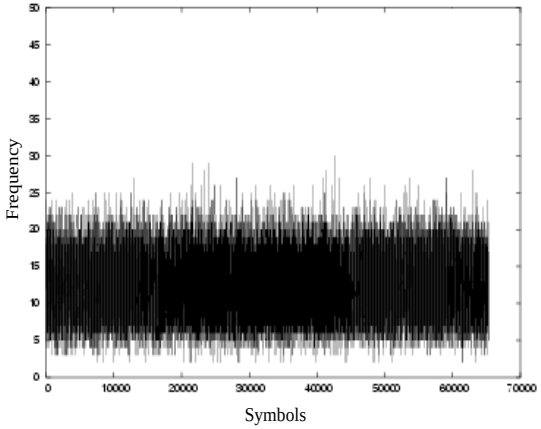

Fig. 3.20: Histogram of 16,00,00 bytes of AES ciphertext resulting from encryption of 16 byte number with 1,00,000 keys.

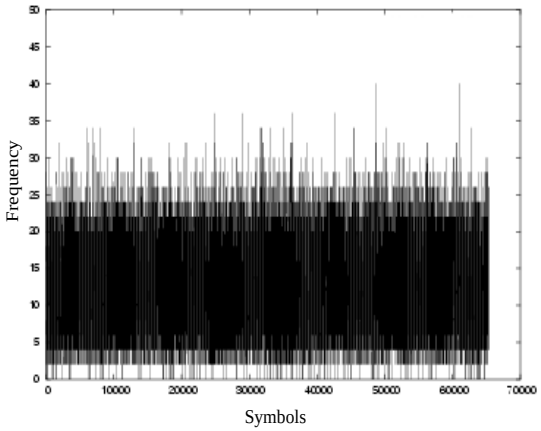

Fig. 3.21: Histogram of 16,00,00 bytes of Blowfish ciphertext resulting from encryption of 16 byte number with 1,00,000 keys.

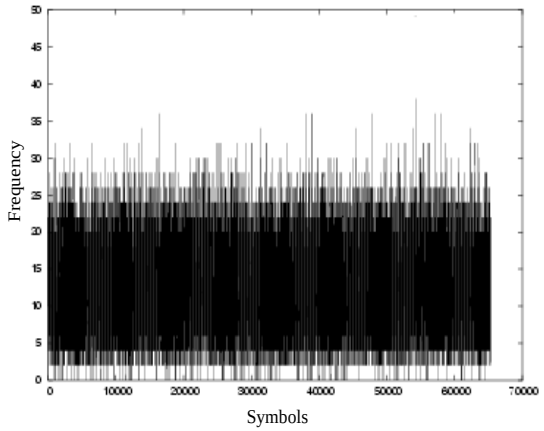

Fig. 3.19: Histogram of 16,00,00 bytes of DES ciphertext resulting from encryption of 16 byte number with 1,00,000 keys.

distributed in the ciphertexts. Entropy of these ciphertexts is very high and it is quite difficult to find the encryption method and to extract the plaintext from the ciphertext. It is observed from histograms that all the five algorithms under consideration DES, AES, Blowfish, TDES, RC5 are strong and do not seem to produce any patterns in the ciphertext with change of key for the same plaintext. Therefore, one may safely assume that these block ciphers with frequent change of key are secure.

3.10 Why ECB ciphertexts are identified better than CBC ciphertexts ? - An analysis

It is observed from the results of all proposed methods for identification of the encryption method that the encryption method can be identified better for the ECB

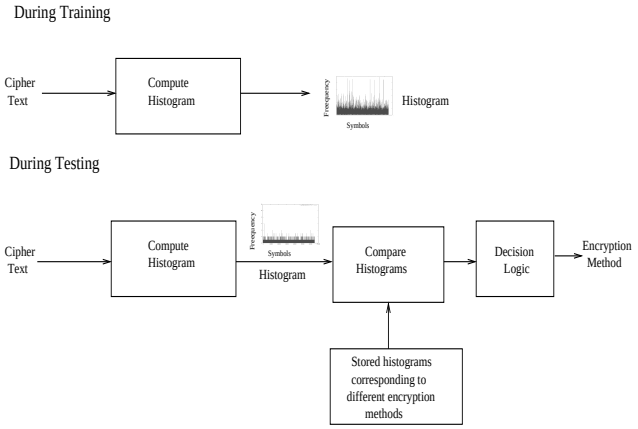

Fig. 3.6: Histogram based classification

Table 3.1: Classification performance (in %) for the histogram based method in identification of the encryption method for block ciphers in ECB mode. Same single key is used in training and testing.

Symbol Size	DES	AES	Blowfish	TDES	RC5	Avg
16bit	100.00	100.00	100.00	100.00	100.00	100.00
12bit	82.50	65.00	90.00	80.00	75.00	78.50
8bit	75.00	7.50	77.50	82.50	77.50	64.00
4bit	35.00	22.50	37.50	40.00	22.50	31.50

Table 3.2: Classification performance (in %) of the histogram based method for CBC mode of encryption. Same single key is used in training and testing.

Symbol Size	DES	AES	Blowfish	TDES	RC5	Avg
16bit	17.50	15.00	22.50	20.00	25.00	20.00
12bit	25.00	15.00	17.50	25.00	15.00	19.50
8bit	22.50	17.50	35.00	22.50	17.50	23.00
4bit	32.50	17.50	30.00	22.50	22.50	25.00

Table 3.3: Classification performance (in %) of the histogram based method with data encrypted using 5 different keys, in ECB and CBC modes, 16-bit symbol size.

Mode of	Encryption Method					Average
Operation	DES	AES	Blowfish	TDES	RC5	
ECB	100.00	100.00	100.00	100.00	100.00	100.00
CBC	38.50	25.00	25.00	18.00	21.50	25.60

same 5 keys is shown in Table 3.3. Here the symbol size is 16 bits. It can be inferred from the results that even combining the ciphertexts that are encrypted with different keys, the patterns still can be captured in the ECB mode, though the performance for the CBC mode data is not much different from the single key case.

Table 3.4 shows the performance of histogram based method for different number of keys. The results indicate that it is possible to identify the encryption method of the ECB mode ciphertexts with 100% accuracy using the histogram method with 16-bit symbol size. In the CBC mode, no distinct patterns are captured by the histogram method from the ciphertexts as the symbols are uniformly distributed.

Table 3.18: Classification performance (in %) for identification encryption method using expansion methods. CBC mode.

Expansion Technique	DES	AES	Blowfish	TDES	RC5	Average
4 to 6 bit expansion	42.50	07.50	05.00	37.50	07.50	20.00
Convolutional Codes	2.50	27.50	25.00	17.50	35.00	22.50

where m/n is the code rate ($n \geq m$) and (b) the transformation is a function of the last k information symbols, where k is the constraint on the length of the code [39]. This method is referred to as the (n, k, m) convolutional code. Viterbi decoding algorithm is used to recover the original binary data [39].

Table 3.18 shows classification results obtained with the help of expansion techniques. Though these techniques are revealing some patterns in the expanded data, these patterns are not very useful, as the same kind of patterns appear in the ciphertexts of all the methods.

3.9 Multiple key attack

In this section we explain a technique to exploit the encryption methods with multiple key attack. In this attack a small plaintext for example credit card number which is a 16 digit long is encrypted with several thousands of keys. The resulting ciphertexts are being observed for any possible patterns.

Here we have taken 16 byte plaintext encrypted with 1,00,000 keys and the resulting 16,00,000 bytes of ciphertext for each encryption method. Figures 3.19, 3.20, 3.21, 3.22, 3.23 show the histograms of the ciphertexts of DES, AES, Blowfish, TDES and RC5 respectively. It is seen from these histograms that the symbols are uniformly

other fields due to [38]. In this technique classification is performed by comparing the compressibility of a given ciphertext to the compressibility of ciphertexts of the known encryption method. The compression techniques like *gzip, bzip2* and *lzma* compress well if the data has a lot of redundancy (low entropy data). Entropy of text data is 8.301079 and for ciphertext it is more than 15.9 for 16 bit symbols. So it is not possible to use compression techniques for identification of the block ciphers as the compression techniques are not able to compress the ciphertexts at all.

3.8.2 Data expansion techniques

Compressed data contains high entropy and it cannot be compressed further, same with the case of ciphertext, it cannot be compressed further. It gives an insight that normally compressed texts do not contain any patterns in it as the symbols in it are uniformly distributed. When it is uncompressed or expanded patterns will be exposed in the data. Here we, try to expand the ciphertext using two methods to capture any distinct patterns that they may produce in the expanded data.

3.8.2.1 4 to 6 bit expansion method

This method is similar to the Expand-Box function in DES. In this method every 4 bits of the ciphertext are expanded to 6 bits. On the whole, the file size will increase by 50%. Using the histogram method, we try to extract the patterns in the expanded data.

3.8.2.2 Convolutional codes method

In telecommunication, a convolutional code is a type of error-correcting code in which (a) each m-bit information symbol to be encoded is transformed into an n-bit symbol,

Table 3.4: Classification performance (in %) of the histogram method in ECB mode for mulitple key ciphertexts. 16-bit symbol size.

Number of keys	Encryption Method					Average
	DES	AES	Blowfish	TDES	RC5	
1	100.00	100.00	100.00	100.00	100.00	100.00
5	100.00	100.00	100.00	100.00	100.00	100.00
20	98.37	97.75	98.37	97.50	98.87	98.17
30	90.91	94.33	95.91	95.00	97.25	94.68

3.2.2 Histogram method-Pairwise classification

The classification strategy in the previous method is the one-against-the-rest, whereas here it is the one-against-one. In this method, we reduce the multi category classification problem to N_{C_2} two category classification problems, where N is the number of categories. We select the nearest known ciphertext class to the test ciphertext in every pair, and finally the encryption method is identified based on the maximum number of votes. Nearest class is selected in every pair based on distinctive features of the two methods. In this technique the distinct properties between any two encryption methods can be clearly observed. These distinct features are used for classification. Classification results for this method are shown in Table 3.5. The results are marginally better than that for the one-against-one strategy as we identify the distinct features between every pair of encryption methods. However these results not satisfactory.

Table 3.5: Classification performance (in %) of the histogram method-pairwise classification of ciphertext of CBC mode. 8bit symbol.

DES	AES	Blowfish	TDES	RC5	Avg
15.00	30.00	37.50	22.50	20.00	25.00

3.2.3 Classification of the first block of CBC mode ciphertext

First block of a ciphertext using the CBC mode of operation is the same as that in the ECB mode, since there is no dependency on the previous blocks [5], except that the initialization vector (IV) is XORed with the first block of the plaintext. If the initialization vector is the same for all the ciphertexts, then it is possible to extract some pattern out of the first blocks of ciphertexts.

Since we are considering only the first block of every ciphertext file, that is 8 bytes in the case of DES and 16 bytes in the case of AES, it requires large number of ciphertexts for computing the histogram with 16-bit symbol size. We used 1,28,000 ciphertexts for training and 51,200 ciphertexts for testing. Training histograms are obtained from the first block of training ciphertexts. In case of testing, one block is not sufficient to compute the Euclidean distance measure. Hence, we used a group of 1280 ciphertexts to construct one histogram for testing. The 51,200 test ciphers are divided into 40 groups, each containing 1280 ciphertexts. The patterns that can be captured in the first block of CBC mode ciphertexts are very much similar to the patterns in the ECB mode ciphertexts. Table 3.6 shows the classification results. The results are on par with the results of the ECB mode ciphertexts. So it is recommended that the IV be changed every time before encryption in the CBC mode to make the ciphers more secure.

Table 3.17: Performance (in %) of the Hamming distance based methods.

Method	DES	AES	Blowfish	TDES	RC5	Avg
Minimum Hamming distance with known ciphertext	29.00	23.00	12.00	21.00	18.00	20.60
Hamming distance from previous ciphertext block	27.00	28.00	30.00	12.00	21.00	23.60

for different encryption methods. In this technique we look at the Hamming distance between consecutive blocks of the ciphertext. It shows how different the next block is from present block. We use the SVMs for modeling the variances. Here the feature vector is the XORed value between consecutive blocks. The feature vector is made up of 1 or -1 as explained in previous section and the size of the feature vector is chosen as the length of the block for the encryption method. Table 3.17 shows the classification results for the Hamming distance based methods. These methods failed in identifying the encryption method, because a small change in the one ciphertext block causes enormous change in the later ciphertext blocks. And also the most unpredictable relationship between consecutive ciphertext blocks leads to poor performance of the Hamming distance based methods.

3.8 Data compression and expansion techniques

3.8.1 Compression techniques

Compression techniques have been used for spoken and written language identification and achieved high success rates [37]. Compression is used for classification in many

Table 3.16: Performance (in %) of the Gaussian mixture models method for identification of the encryption method in ECB and CBC modes.

Mode of Operation	Encryption Method					Average
	DES	AES	Blowfish	TDES	RC5	
ECB	70.00	70.00	67.50	22.50	57.50	57.50
CBC	20.00	15.00	47.50	07.50	05.00	19.00

3.7 Hamming distance based methods

The Hamming distance between two strings of equal length is the number of positions for which the corresponding symbols are different. In other words, it measures the minimum number of substitutions required to change one string to the other. Encryption algorithms perform many bitwise operations on the plaintext to transform it to the ciphertext. So Hamming distance is a measure of the difference between two ciphertext blocks. XOR operation gives bit difference between two data blocks.

3.7.1 Minimum Hamming distance with known ciphertext

In this technique we compute the Hamming distance of a given ciphertext and the training ciphertext data of each encryption method. The encryption method of a given ciphertext can be determined using the minimum Hamming distance with training samples.

3.7.2 Hamming distance from previous ciphertext block

In the CBC mode each ciphertext block depends on its preceding block. We assume that the variance between consecutive CBC mode ciphertext blocks would be different

Table 3.6: Classification performance (in %) based on the histogram for the first-block of ciphertext for CBC mode. The symbol size is 16 bits. Initialization vector is constant

DES	AES	Blowfish	TDES	RC5	Avg
100.00	100.00	100.00	100.00	100.00	100.00

Table 3.7: Entropy for different types of plaintext data for symbol size of 16bits.

Text Data	Speech Data	Image Data
8.3010	13.8970	11.7536

3.2.4 Identification of encryption method for ciphertexts of image and speech data

Like the textual data, multimedia data such as speech and image data also have low entropy, and have so much variations of frequency in their histograms. These variations would reflect in the ciphertext causing high success rate in identifying the encryption method. Figures 3.7, 3.8 and 3.9 show the histograms of the textual, speech and image data respectively. It is evident from these histograms that they have less entropy which enables high classification rate. Entropy of textual, speech and image is shown in the Table 3.7. The results of identification of the encryption method on image and speech data are given in Table 3.8. The results are similar to that of text data.

3.2.5 Identification of encryption mode

Identification of the mode of encryption is important because, without knowing the mode of encryption it is not possible to proceed with the cryptanalysis. The Histogram

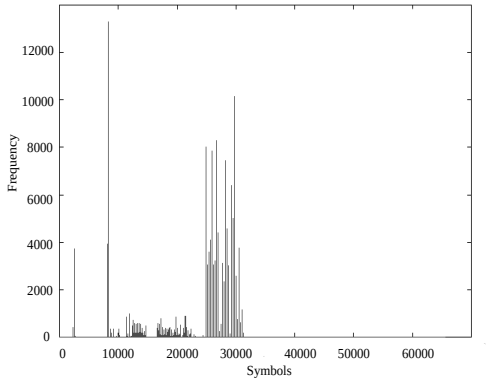

Fig. 3.7: Histogram of textual data.

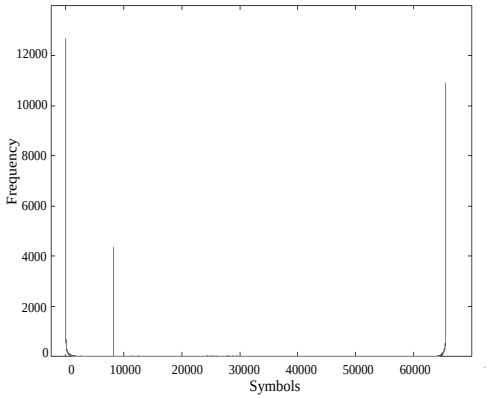

Fig. 3.8: Histogram of speech data.

through expectation-maximization algorithm [33]. In this respect, the GMMs are very similar to support vector machines and neural networks, and all of these models have been used in various fields such as speech recognition [34], speaker verification [35] and image classification [36] fields. Success in these fields with GMMs prompted us to apply this method of classification for classifying the ciphertexts based their encryption method.

A GMM allows distribution of the observed parameters, represented by m mixture Gaussian components in the form of

$$p(X) = \sum_{i=1}^{m} \alpha_i N(X, \mu_i, \Sigma_i) \tag{3.7}$$

Where α_i denotes the priority probability component of i, ($\sum_{i=1}^{m} \alpha_i = 1$ and $\alpha_i \geq 0$) and $N(X, \mu_i, \Sigma_i)$ denotes the normal distribution of the i^{th} component with the mean vector μ_i and covariance matrix Σ_i in the form of

$$N(X, \mu_i, \Sigma_i) = \frac{1}{(2\pi)^{p/2}\Sigma_i^{1/2}} \exp[-\frac{1}{2}(X - \mu_i)^T {\Sigma_i}^{-1}(X - \mu_i)] \tag{3.8}$$

where p is the number of dimensions. The parameters of the model (X, μ, Σ) can be estimated using expectation maximization algorithm(EM). Models for each encryption method are built using the training data . The feature vector is made up of 1 or -1 as explained in the previous section and the size of the feature vector is same as the block size for the encryption method. Table 3.16 shows the classification results of GMM method for identification of encryption method of ciphertexts. The GMM failed to separate observed data into components as the ciphertext data is quite randomly distributed. Hence, the results show poor performance of GMM method.

Table 3.14: Classification performance (in %) for block prediction method in bitwise representation, for different input block sizes. Same key used in training and testing

Input Blocks	DES	AES	Blowfish	Triple DES	RC5	Average
1	17.50	22.50	20.00	15.00	10.00	17.00
3	20.00	17.50	12.50	25.00	22.50	19.50
4	22.50	25.00	27.50	10.00	25.00	22.00

Table 3.15: Classification performance (in %) for block prediction method in bit group representation, for different input block sizes. Same key used in training and testing

Input Blocks	DES	AES	Blowfish	Triple DES	RC5	Average
1	15.00	22.50	22.50	20.00	25.00	21.00
3	12.50	20.00	20.00	15.00	22.50	18.00
4	15.00	27.50	12.50	17.50	25.00	19.50

level of nonlinearity of the CBC mode ciphertext data. The dependence of a CBC mode ciphertext block on the previous blocks is quite unpredictable and it can not be captured by the pattern recognition methods.

3.6 Gaussian mixture model method

In this technique we consider the use of Gaussian mixture model(GMM) [32] technique for identifying the possible patterns in the ciphertexts of both the ECB and CBC modes. The GMMs are a class of pattern recognition techniques which model the probability density function of observed variables using a multivariate Gaussian mixture density. Given a series of inputs, it refines the weights of each distribution

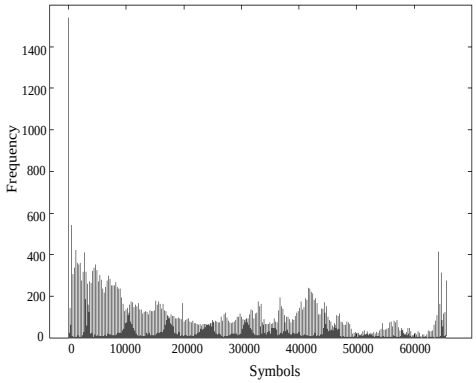

Fig. 3.9: Histogram of Image data.

Table 3.8: Classification rate (in %) of textual, speech and Image ciphertexts. 16bit symbol. ECB Mode

Data Type	DES	AES	Blowfish	TDES	RC5	Avg
Text	100.00	100.00	100.00	100.00	100.00	100.00
Speech	100.00	100.00	97.50	100.00	100.00	99.50
Image	80.00	97.50	85.00	90.00	82.50	87.00

Table 3.9: Entropy for different modes of ciphertexts for a symbol size of 16-bits.

Encryption Method	CBC Mode	ECB Mode
DES	15.904717	15.785736
AES	15.905167	15.903238
Blowfish	15.904456	15.782907
TDES	15.904639	15.783682
RC5	15.904633	15.783822

technique can also be used for identifying the encryption mode in block ciphers. Figures 3.1, 3.2, 3.3, 3.4 and 3.5 show the histograms for ciphertexts obtained using DES, AES, Blowfish, TDES and RC5 encryption methods in the ECB and CBC modes. These histograms clearly show the difference between the CBC mode and ECB mode of encryption. In ECB mode, the variations in frequency of occurrence of symbols are high compared to the variations in frequency in the CBC mode. Empirical results show that it is possible to identify the mode of encryption of any ciphertext with 100% accuracy based on the entropy of symbols in a ciphertext for DES, Blowfish, TripleDES and RC5 encryption methods. Table 3.9 shows the entropy of the symbols in the ciphertext. Entropy of the symbols in the plaintext is 8.056235 for 16-bit symbol size. There is a considerable difference in the entropy between the ECB mode and CBC mode ciphertexts except for the AES. Entropy of the CBC mode ciphertexts is around 15.9 whereas for the ECB mode ciphertexts it is around 15.78 except for the AES. Therefore the encryption mode for a given ciphertext is determined using the entropy of the ciphertexts.

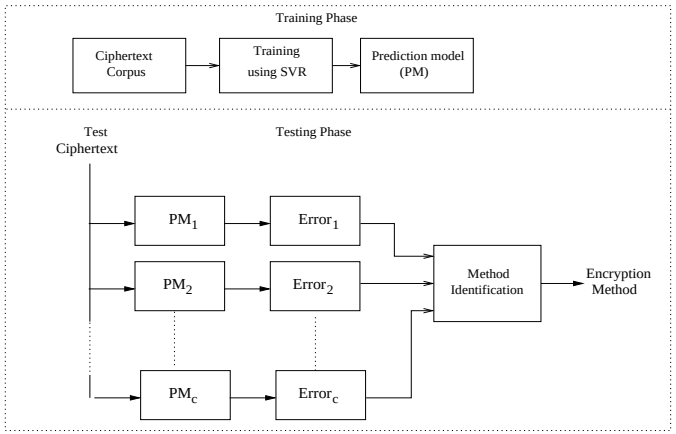

Fig. 3.18: Identification of the encryption method using block prediction method.

great extent. Bitwise representation of ciphertext is shown below with 3 input blocks.

$$b_1^1, b_2^1..b_{64}^1, b_1^2...b_{64}^2, b_1^3....b_{64}^3, \hat{b}_k \tag{3.6}$$

here b_j^i represents j^{th} bit of i^{th} block, and $\hat{b}_k$ represents k^{th} bit of the block to be predicted.

Another approach for implementing the block prediction method is considering k-bit groups of the ciphertext as unsigned integer or normalized integer values. In our studies we have studied for three different values of k (4, 8 and 16). The number of input blocks can vary from 1 to 8, but the empirical results show that increasing the number of input blocks does not improve the classification rate.

Table 3.14 shows the classification performance using the block prediction method using bitwise representation and Table 3.14 using bit group representation of size 8bits in each group. It can be inferred from the results that the support vector regression is not able to predict the next possible block of ciphertext. It indicates the high

3.5.2 Building block prediction models

A block prediction method predicts *(l+1)*th block, given the previous l input blocks. Model should be built in such a way that if the ciphertext contains k blocks of data, k-l blocks are predicted, for every consecutive l blocks the next block is predicted. The following example illustrates the mechanism. Consider the ciphertext given below

$$B_1, B_2, B_3, B_4, B_5....B_l, B_{l+1}, ..., B_k \tag{3.5}$$

where $B_1, B_2...B_k$ are 64bit blocks in a ciphertext (block size varies with the encryption method, for DES it is 64 bit). Given $B_{p-l}, B_{p-l+1}, ...B_p$, ${B_{p+1}}^{th}$ block is predicted. The previous l blocks are used to predict the *(l+1)*th block. The process is repeated for all consecutive l blocks in ciphertext, and the corresponding $(k-l)$ predicted blocks are obtained. Using the SVR, prediction models for each encryption method are built and are used later for testing a given test ciphertext.

Figure 3.18 shows the process of identification of the encryption method using the block prediction method. A classifier is constructed using the prediction models of all encryption methods. For any given ciphertext, the error is calculated as the Euclidean distance between the predicted ciphertext data and the actual ciphertext data. The encryption method is identified as the one whose model gives the least error.

3.5.3 Implementation of block prediction method

Block prediction method can be implemented in two ways based on the representation of ciphertext namely, bitwise representation and bit-group representation. In bitwise representation a ciphertext block of length k bits is represented as k numbers of either 1 or -1. It is similar to the binary representation of block except that -1 is used instead of zero for better prediction and analysis. The number of input blocks used for the prediction plays a major role in this technique which influences classification rate to a

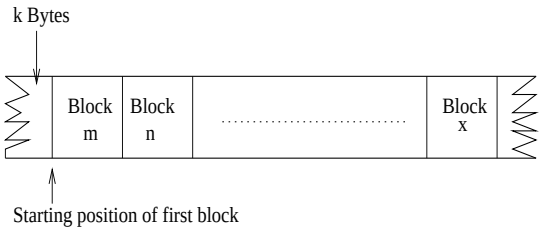

Fig. 3.10: Missing initial bytes in the captured ciphertext.

3.2.6 Identification of the starting position of ciphertext block

Often, the captured ciphertexts are not available from the starting position of actual data, as some of the initial bytes in it may be missing or only a chunk of actual data is obtained from the source as shown in Figure 3.10. In this case it is not possible to identify the encryption method of the ciphertext, and analysis becomes complex and therefore decryption is also not possible. It will be useful if we can find the starting position of the block for a given ciphertext or in other words we should identify how many bytes are missing in the first block of a given ciphertext. So that we analyze the ciphertext from the proper next block onwards. The approach given below finds the starting position of the first proper ciphertext block with nearly 100% accuracy for the ECB mode.

Let the block size be 64 bits. We use the training ciphertexts of an encryption method to compute histogram for each of the 8 byte positions in a block. For example, we compute the first histogram for the first byte of all the blocks in the training data set. Let this reference histogram be called RH_1. In a similar way, the histograms for the other seven byte positions in a block are computed and are called as RH_2, RH_3, RH_4, RH_5, RH_6, RH_7, and RH_8. These reference histograms computed from

the DES training data set are shown in Figure 3.11. It is seen that histograms for different byte positions are different. A feature vector for training data, Z_R is formed by concatenating the 8 histograms, RH_1, RH_2 and so on till RH_8, in that order.

For a given test ciphertext data, the 8 histograms as in the case of training data are computed. Let the test histograms be called as TH_1, TH_2, and so on till TH_8. The histograms computed from a test ciphertext data are shown in Figure 3.12. Then 8 feature vectors are formed from the test data histograms as follows: The feature vector Z_{T_1} is formed by concatenation of TH_1, TH_2, and so on till TH_8, in that order. The feature vector Z_{T_2} is formed by concatenation of TH_2, TH_3, TH_4, TH_5, TH_6, TH_7, TH_8, TH_1, in that order. In a similar way, the other 6 feature vectors Z_{T_3}, Z_{T_4}, Z_{T_5}, Z_{T_6}, Z_{T_7}, Z_{T_8} are formed by rotating one element each time using the test data histograms.

$$
\begin{aligned}
Z_R &= \{RH_1, RH_2, \cdots, RH_8\} && (3.1)\\
Z_{T_1} &= \{TH_1, TH_2, \cdots, TH_8\} \\
Z_{T_2} &= \{TH_2, TH_3, \cdots, TH_8, TH_1\} \\
Z_{T_3} &= \{TH_3, TH_4, \cdots, TH_8, TH_1, TH_2\} \\
&\vdots \\
Z_{T_8} &= \{TH_8, TH_1, TH_2, \cdots, TH_7\} && (3.2)
\end{aligned}
$$

The training data feature vector that has the minimum Euclidean distance with the test data feature vector is determined as follows:

$$N = \arg\min_i \mid\mid Z_R - Z_{T_i} \mid\mid^2 \tag{3.3}$$

The value of N gives the position of the starting byte of a block in the test ciphertext data. By removing $N-1$ initial bytes, we can reach the starting position of the

The CBC mode ciphers being highly non-linear, this is an appropriate technique to capture the relation between a ciphertext block and its previous ciphertext blocks. In the following section we give a brief overview of the support vector machines(SVM) and regression based on SVM.

3.5.1 Support vector machines

Support vector machine(SVM) is a training algorithm for learning the classification rules from the data. SVMs non-linearly map their n-dimensional input space into a high dimensional feature space. In this high dimensional feature space a linear classifier is constructed [30]. SVM can be used for regression also. Support vector regression(SVR) is a powerful technique for predictive data analysis. In a regression model the dependence of a scalar d on a vector x is described by

$$d = f(x) + r \tag{3.4}$$

For a given set of training data the problem is to provide an estimate of the dependence of d on x. In support vector regression, nonlinear regression is considered as linear regression by transforming data into space onto the high dimensional feature space [30].

Given the training data $(x_1, y_1), ...(x_k, y_k)$ subset of $X \in R$ where X denotes space of input patterns, for instance R_d. In SVR, our goal is to find $f(x)$ that has at most ϵ deviation from the actually obtained target y_i, for all training data. In other words we don't care about errors as long as they are less than ϵ, but will not accept deviation larger than this. SVR has been used and has yielded excellent results in time series prediction [31]. In block prediction method SVR is used to model the series of ciphertext blocks and to predict the next ciphertext block.

Table 3.13: standard deviation of probability of bit '1' in all the bit positions of a ciphertext block of ECB and CBC modes.

Encryption Method	ECB	CBC
DES	0.003305	0.002201
AES	0.001771	0.001645
Blowfish	0.003001	0.001822
TDES	0.003102	0.002122
RC5	0.002389	0.001764

bit positions of a ciphertext block. The results indicate that the block ciphers under consideration are distributing the bits equally in the entire block. So the attacks based on bit position cannot succeed.

3.5 Block Prediction Method

In this section we present a detailed description of the block prediction method and its implementation. Block prediction method is mainly used for identifying the encryption method of a ciphertext encrypted in the CBC mode. This technique tries to exploit the chaining property of the CBC mode.

Given a series of l input blocks of a ciphertext, the method tries to predict the $(l+1)^{th}$ block. It is a statistical measure that attempts to determine the strength of the relationship between one block and a series of input blocks preceding it. The error corresponding to the difference between the predicted block and actual block is used for classification. Support vector regression technique(SVR) [29] is used to predict the $(l+1)^{th}$ block. SVR can capture non-linear relationship among series of values.

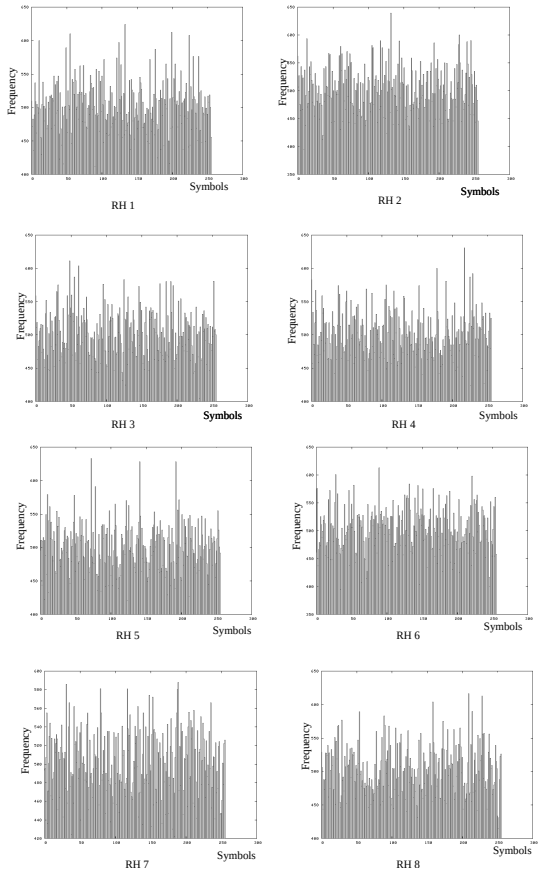

Fig. 3.11: Histograms for different byte positions in a block of DES training data ciphertexts.

Table 3.10: Performance (in %) of the multiple histograms method for identifying the starting position of block in ECB and CBC modes. Same key is used in training and testing

Mode of	Encryption Method				
Operation	DES	AES	Blowfish	TDES	RC5
ECB	100.00	20.00	100.00	100.00	100.00
CBC	12.50	15.00	10.00	20.00	17.50

next proper block in the captured ciphertext.

Table 3.10 shows the performance of this method in identifying starting position in both ECB and CBC modes. Results show that the starting position of a block in the ECB mode can be identified with nearly 100% accuracy for all the methods except for AES, whereas in the CBC mode the results are very poor. Poor performance in identifying the starting position in the AES is attributed to its fairly uniform distribution of symbols.

3.3 Multiple Histograms Method

In this technique, we take multiple histograms of the ciphertext instead of just one, based on the positions of the ciphertext bits in every block. The motivation for this method is to reduce the entropy of the data we are examining. So that we may capture patterns in the data. Implementation of this method is similar to the method given in Section 3.2.6. Reference histograms are computed from the training ciphertexts, one for each byte position. Given a ciphertext, multiple histograms are computed similar to training data, and these histograms are compared with the reference histograms to identify the encryption method. Table 3.11 shows the classification results for this

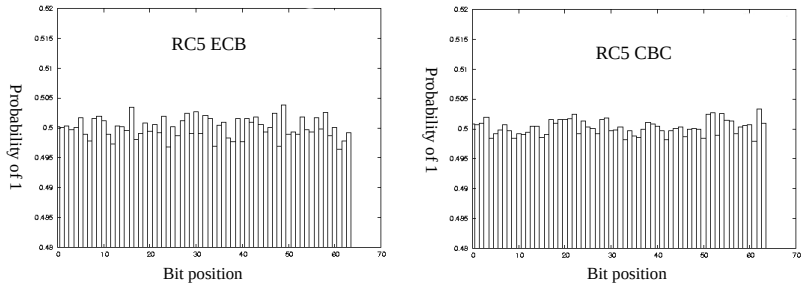

Fig. 3.17: Histograms probability of '1' in different bit positions in a block of RC5 ciphertext.

Table 3.12: Performance (in %) of the Bitwise histograms method for identification of encryption algorithm in ECB and CBC modes. Same key used in training and testing

Mode of	Encryption Method					Average
Operation	DES	AES	Blowfish	TDES	RC5	
ECB	45.00	17.50	30.00	27.50	45.00	33.00
CBC	15.00	22.50	10.00	27.50	22.50	19.50

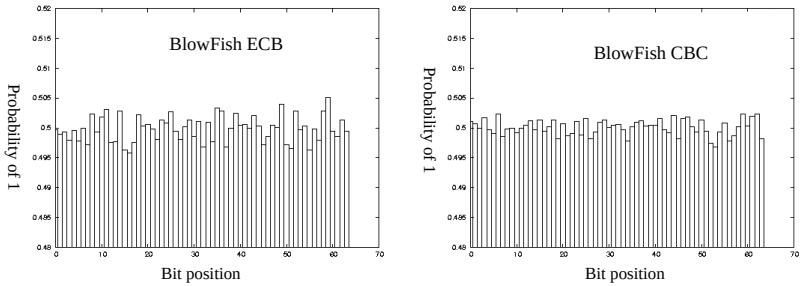

Fig. 3.15: Histograms probability of '1' in different bit positions in a block of Blowfish ciphertext.

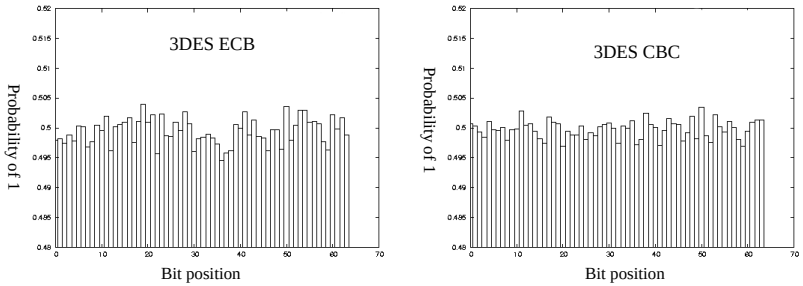

Fig. 3.16: Histograms probability of '1' in different bit positions in a block of TDES ciphertext.

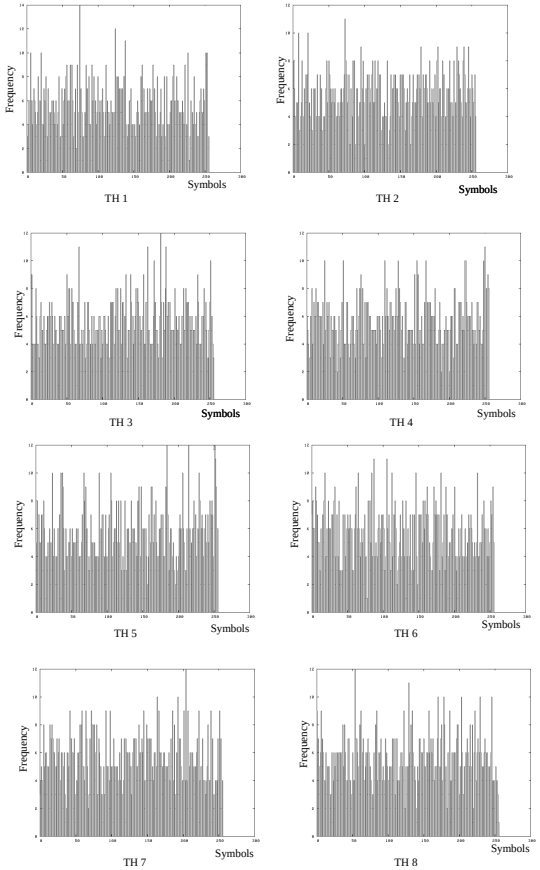

Fig. 3.12: Histograms for different byte positions in a block of a DES test data ciphertext.

Table 3.11: Performance (in %) of the Multiple histograms method for identification of encryption algorithm in ECB and CBC modes. Same key used in training and testing

Mode of	Encryption Method				
Operation	DES	AES	Blowfish	TDES	RC5
ECB	100.00	100.00	100.00	100.00	100.00
CBC	17.50	25.00	22.50	20.00	22.50

method. Multiple histogram method reduces the entropy of the observed data by dividing the ciphertext into multiple bins in the ECB mode. In case of the CBC mode the entropy remained same and therefore the results are no different from the earlier histogram based methods. The CBC mode ciphertexts are highly nonlinear and have very high entropy. Dividing the ciphertexts into different bins based on bit positions does not reduce the entropy. Hence resulting in poor performance in the encryption method identification.

3.4 Bitwise Histograms

In this section we discuss the technique based on bitwise histograms. It deals with the probability that a bit is set at a particular position in a ciphertext block. This technique is designed to capture the probability of occurrence of '1' in each bit position of the ciphertext block. This information can be used for classifying the ciphertext. The frequency of occurrence of 1 in each bit position in all the blocks are observed here. Figures 3.13, 3.14, 3.15, 3.16 and 3.17 show the probability of occurrence bit 1 in bit positions from 1 to 64 for DES, AES, Blowfish, TDES and RC5 respectively. It is seen from the histograms that probability of 1 is almost the same for all bit positions

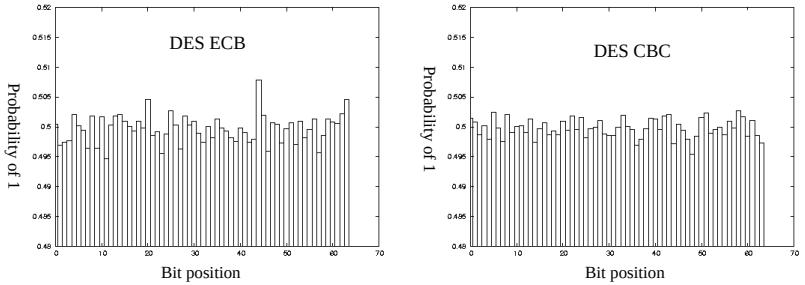

Fig. 3.13: Histograms probability of '1' in different bit positions in a block of DES ciphertext.

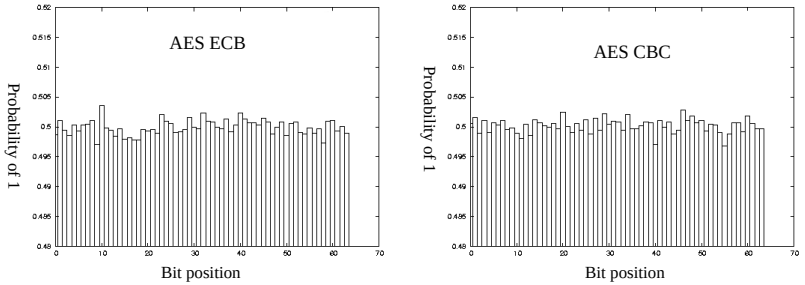

Fig. 3.14: Histograms probability of '1' in different bit positions in a block of AES ciphertext.

in the CBC mode while there is slight variation in the ECB mode. Hence the results are poor for this method. Table 3.12 shows the classification results.

Contrary to our assumption bitwise histogram method does not perform as expected. The reason is that the probability of occurrence of 1 in every bit position of ciphertext block for all methods is very near to 0.5 and the standard deviation is very small. Table 3.13 shows the standard deviation of probability of bit '1' in the all the