# Cryptographic Techniques

Soumyadip Mal[1], Utsab Banerjee[2]
[1]B.Tech 4th year, [2]B.Tech 4th year
Department of Electrical Engineering, Netaji Subhash Engineering College,Kolkata,India

***Abstract*—In today's age, one which is witnessing a burst of advancement of technology, like never, security and privacy of information and communication has become a very important aspect. Herein lies the advantages of cryptography and cryptanalysis. Cryptography is a way that makes sure of the integrity, availability and identification, confidentiality, authentication of user and as well as security and privacy of data that can be provided to the user. In this paper we have defined and analyzed various symmetric cryptographic algorithms like DES, Triple DES, Blowfish, AES and IDEA and asymmetric key cryptographic algorithms like RSA. They have been analyzed on their ability to secure data, key size, block size, features. We have also forayed into the realms of DNA Cryptography, Elliptic Curves based Cryptography as well as Quantum Cryptography all of which are emerging trends in the domain of Cryptography but hold massive potential nonetheless, we believe.**

***Keywords*-- Cryptography, Encryption, DES, Diffie Hellman, RC5, Triple DES, AES, RSA,Quantum Cryptography,qubits,DNA Cryptography,Elliptic Curve Cryptography,Chaotic Cryptography,Shannon's diffusion.**

## I. INTRODUCTION

**Cryptography** or **cryptology** is derived from Greek word *kryptós* meaning "hidden, secret" and *graphein*, meaning "writing" or "study"[1][2].It is the study of the science and art behind securing communication from any third parties or the public.Modern cryptography is mostly based on mathematical theory and computer science practice; cryptographic algorithms are designed around computational hardness assumptions, making such algorithms hard to break in practice by any adversary. It is theoretically possible to break such a system, but it is infeasible to do so by any known practical means. These schemes are therefore termed computationally secure; theoretical advances, e.g., improvements in integer factorization algorithms, and faster computing technology require these solutions to be continually adapted. There exist information-theoretically secure schemes that provably cannot be broken even with unlimited computing power—an example is the one-time pad—but these schemes are more difficult to implement than the best theoretically breakable but computationally secure mechanisms.There have been classical methods of Crypptography which have been around for some time now.But the more advanced ones have all been invented in the computer era.Now,let us breakdown a few jargons related to Cryptography before we get into it.

**Cryptography Terminologies**[2]**:**

- **Plain Text**: Any communication in the language that we speak that is the human language, takes the form of plain text. It is understood by the sender, the recipient and also by anyone who gets an access to that message.
- **Cipher Text**: Cipher means a code or a secret message. When a plain text is codified using any suitable scheme the resulting message is called as cipher text.
- **Encryption**: The process of encoding plain text messages into cipher text messages is called encryption.
- **Decryption**: The reverse process of transforming cipher text messages back to plain text is called as decryption.
- **Key**: An important aspect of performing encryption and decryption is the key. It is the key used for encryption and decryption that makes the process of cryptography secure.

**Purpose of Cryptography:**

Cryptography serves following purposes[3]:

- **Confidentiality**: The principle of confidentiality specifies that only the sender and the intended recipient should be able to access the contents of a message.
- **Authentication**: Authentication mechanisms help to establish proof of identities. This process ensures that the origin of the message is correctly identified.
- **Integrity**: The integrity mechanism ensures that the contents of the message remain the same when it reaches the intended recipient as sent by the sender.
- **Non-repudiation**: Non-repudiation does not allow the sender of a message to refute the claim of not sending the message.
- **Access Control**: Access Control specifies and controls who can access what.
- **Availability**: The principle of availability states that resources should be available to authorized parties all the times.

## II. TYPES OF CRYPTOGRAPHY

There are mainly two types of cryptography[3][1]:

**1) Secret key cryptography or Symmetric-key cryptography**:

In SKC,the sender and the receiver know the same secret code, which is known as key. With the same key messages are encrypted by the sender and decrypted by the receiver. It can be of 2 types : Stream Buffer, Block Buffer.Stream Buffer: Stream buffer encrypts the digits of a message one at a time. Stream Cipher functions is used on a stream of data one at time by operating on it by bits. It consists of two components: 1) a key stream generator and 2) mixing function. Mixing function uses XOR function, and key stream generator is unit in stream encryption algorithm.
Block cipher : In Block cipher, it takes a number of bits and then encrypt them as a single unit. Data is encrypted/decrypted if data is in the forms of blocks. In simple words , the plain text is divided into blocks which are used to produce blocks of cipher textpadding the plaintext in blocks. 64 bits blocks have been commonly used.

**2) Public key cryptography or Asymmetric-key cryptography:**
Asymmetric key (or public key) encryption is used to solve the problem of key distribution. In PKC, two keys are used; private keys and public keys. For encryptionpublic key is used and for decryption private key is used . Public key is known to public and private key is known to the user.

## III. CRYPTOGRAPHY ALGORITHMS:

The various cryptography algorithms are as follows :
**DES**: DES[3][15][11] is a block cipher that uses shared secret key for encryption and decryption. DES algorithm as described by Davis R takes a fixed-length string of plaintext bits and transforms it through a series of complicated operations into cipher text bit string of the same length. In the case of DES, each block sizeis 64 bits. DES also uses a keyof 56 bits to customize the transformation, so that decryption can only be performed by those who know the particular key used to encrypt the message. There are 16 identical stages of processing,termed rounds. There is also an initial and final permutation, termed IPand FP, which are inverses(IP "undoes" the action of FP, and vice versa). The Broad level steps in DES are as follows :
1. In the first step, the 64-bit plain text message is handed over to an Initial permutation (IP) function.
2 .The initial permutation is performed on plain text.
3. The IP produces two halves of the permuted message; Left Plain Text (LPT) and Right Plain Text (RPT).
4.. Now, each of LPT and RPT go through 16 rounds of encryption process.
5.In the end, LPT and RPT are rejoined and a final permutation (FP) is performed on the combined block.
6.The result of this process produces 64-bit cipher text.Rounds: Each of the 16 rounds, in turn, consists of the broad level steps and shownin Figure 3.1.

**Triple DES (3DES):** 3DES or the Triple Data Encryption Algorithm (TDEA)[3][15][11] was developed to address the obvious flaws in DES without designing a whole new cryptosystem. Data Encryption Standard (DES) uses a 56-bit key and is not deemed sufficient to encrypt sensitive data. 3-DES simply extends the key size of DES by applying the algorithm three times in succession with three different keys. The combined key size is thus 168 bits (3 times 56). TDEA involves using three 64-bit DEA keys (K1, K2, K3) in Encrypt-Decrypt-Encrypt (EDE) mode, that is, the plain text is encrypted with K1, then decrypted with K2, and then encrypted again with K3 [17]. The standards define three keying options:
Option 1, the preferred option, employs three mutually independent keys ($K1 \neq K2 \neq K3 \neq K1$). It gives keyspace of $3 \times 56 = 168$ bits .Option 2 employs two mutually independent keys and a third key that is the same as the first key ($K1 \neq K2$ and $K3 = K1$). This gives key space of $2 \times 56 = 112$ bits
.Option 3 is a key bundle of three identical keys ($K1 = K2 = K3$). This option is equivalent to DES Algorithm.In 3-DES the 3-times iteration is applied to increase the encryption level and average time. It is a known fact that 3DES is slower than other block cipher methods.

**Advanced Encryption Standard (AES):** Advanced Encryption Standard (AES)[3][15][11] algorithm not only for security but also for great speed. Both hardware and software implementation are faster still. New encryption standard recommended by NIST to replace DES. Encrypts data blocks of 128 bits in 10, 12 and 14 round depending on key size as shown in Figure -2. It can be implemented on various platforms specially in small devices. It is carefully tested for many security applications.
**Algorithm Steps**:
i) **These steps used to encrypt128-bit block**

1. The set of round keys from the cipher key.
2. Initialize state array and add the initial round key to the starting state array.
3. Perform round= 1 to 9 : Execute Usual Round.
4. Execute Final Round.
5. Corresponding cipher text chunk output of Final Round Step

**ii) Usual Round:Execute** the following operations which are described above.

1. Sub Bytes
2. Shift Rows
3. Mix Columns
4. Add Round Key , using K(round)

**iii) Final Round**: Execute the following operations which are described above.

1. Sub Bytes
2. Shift Rows
3. Add Round Key, using K(10)

**iv) Encryption** :Each round consists of the following four steps:

1. Sub Bytes : The first transformation, Sub Bytes, is used at the encryption site. To substitute a byte, we interpret the byte as two hexadecimal digits.
2. Shift Rows : In the encryption, the transformation is called Shift Rows.

3. Mix Columns : The Mix Columns transformation operates at the column level; it transforms each column of the state to a new column.
4. Add Round Key : Add Round Key proceeds one column at a time. Add Round Key adds a round key word with each state column matrix; the operation in Add Round Key is matrix addition.

The last step consists of XO Ring the output of the previous three steps with four words from the key schedule. And the last round for encryption does not involve the "Mix columns" step.

**v) Decryption**: Decryption involves reversing all the steps taken in encryption using inverse functions like

a) Inverse shift rows
b) Inverse substitute bytes
c) Add round key and
d) Inverse mix columns.

The third step consists of XORing the output of the previous two steps with four words from the key schedule. And the last round for decryption does not involve the "Inversemix columns" step.

**Blowfish:** Blowfish is one of the most common public domain encryption algorithms provided by Bruce Schneier -one of the world's leading cryptologists, and the president of Counterpane Systems, a consulting firm specializing in cryptography and computer security.The Blowfish algorithm was first introduced in 1993.The blowfish encryption is shown in figure below:
**Operation of Blowfish:** Blowfish encrypts 64-bit block cipher with variable length key.It contains two parts

- Subkey Generation: This process converts the key upto 448 bits long to subkeys to totaling 4168 bits.
- Data Encryption: This process involves the iteration of a simple function 16 times. Each round contains a key dependent permutation and key-and data dependent substitution.

Blowfish suits the applications where the key remain onstant for a long time (e.g. communication link encryption) but not where the key changes frequently (e.g.
packet switching).

**IDEA(International Data Encryption Algorithm):** DEA[3][15][11] is a block cipher algorithm and it operates on 64-bit plaintext blocks. The key size is 128 bits long. The design of algorithm is one of mixing operations from different algebraic groups. Three algebraic groups are mixed, and they are easily implemented in both hardware and software: XOR, Addition modulo 216 , Multiplication modulo 216 + 1. All these operations operate on 16-bit sub-blocks. This algorithm is efficient on 16-bit processors. IDEA is symmetric key algorithm based on the concept of Substitution-Permutation Structure, is a block cipher that uses a 64bit plain text with 8 rounds and a Key Length of 128-bit permuted into 52 sub-keys each of 128-bits. It does not contain S-boxes and same algorithm is used in reversed for decryption .
**RC4:** RC4 is a stream cipher symmetric key algorithm. as the data stream is simply XOR with generated key sequence. It uses a variable length key 256 bits to initialize a 256-bit state table. A state table is used for generation of pseudo-random bits which is XOR with the plaintext to generate the cipher text.
**RC6:** RC6 is a derivative of RC5. RC6 is designed by Matt Robshaw, Ron Rivest Ray Sidney and is a symmetric key algorithm that is used to congregate the requirements of AES contest . RC6 was also presentedto the CRYPTREC and NESSIE projects. It is patented by RSA Security . RC6 offers good performance in terms of security and compatibility. RC6 is a Feistel Structured private key algorithm that makes use a 128 bit plain text with 20 rounds and avariable Key Length of 128, 192, and 256 bit. As RC6 works on the principle of RC that can sustain an extensive range of key sizes, word-lengths and number of rounds, RC6 does not contain S-boxes and same algorithm is used in reversed for decryption.
**Serpent:** Serpent is an Advanced Encryption Standard (AES) competition, stood 2nd to Rijndael, is a symmetric key block cipher, designed by Eli Biham, Ross Anderson, and Lars Knudsen. Serpent is a symmetric key algorithm that is based on substitution-permutation network Structure. It consists of a 128 bit plain text with 32 rounds and a variable Key Length of 128, 192 and 256 bit. It also contains 8 S-boxes and same algorithm is used in reversed for decryption. Security presented by Serpent was based on more conventional approaches than the other AES finalists. The Serpent is open in the public sphere and not yet patented.
**Twofish:** Twofish is also a symmetric key algorithm based on the Feistel Structure and was designed by Bruce Schneier along with Doug Whiting, John Kelsey, David Wagner, Niels Ferguson and Chris Hall,. The AES is a block cipher that uses a 128 bit plain text with 16 rounds and a variable Key Length of 128, 192, 256 bit. It makes use of 4 S-boxes (depending on Key) and same

algorithm is used in reversed for decryption. The inventors extends the Blowfish team to enhance the earlier block cipher Blowfish to its modified version named Twofish to met the standards of AES for algorithm designing. It was one of the finalists of the AES , but was not selected for standardization. The Twofish is an open to public sphere and not yet patented.

**TEA:** TEA is also a Feistel Structured symmetric key algorithm. TEA is a block cipher that uses a 64 bit plain text with 64 rounds and a Key Length of 128-bit with variable rounds having 32 cycles. It does not contain S-boxes and same algorithm is used in reversed for decryption. TEA is designed to maximize speed and minimize memory footprint. Cryptographers have discovered three related-key attacks on TEA. Each TEA key can be found to have three equal keys, thus it can be used as a hash function. David Wheeler and Roger Needham have proposed extensions of TEA that counter the above attacks.

**CAST:** CAST is symmetric key algorithm based on the backbone concept of Feistel Structure. It is designed by Stafford Taveres and Carlisle Adams, is considered to be a solid algorithm. The CAST is a block cipher that uses a 64 bit plain text with 12 or 16 rounds and a variable Key Length of 40 to128-bit. It also contains 4 S-boxes and same algorithm is used in reversed for decryption. Bruce Schneier, John Kelsey, and David Wagner have discovered a related-key attack on the 64 bit of CAST that requires 2^17 chosen plaintexts, one related query, and 2^48 offline computations. CAST is patented, which was generously released it for free use.

**RC2:** RC2 is designed by Ron Rivest and a variable-key-size encryption algorithm from 0 bytes to the maximum string length that the computer system supports. RC2 is a variable-key-size 64-bit block cipher. It is designed to be a replacement for DES. RC2 is three times faster than DES in software implementations. The algorithm encryption speed is independent of key size.

**RSA:** RSA is designed by Ron Rivest, Adi Shamir, and Leonard Adleman in 1978. It is one of the best known public key cryptosystems for key exchange or digital signatures or encryption of blocks of data. RSA uses a variable size encryption block and a variable size key. It is an asymmetric (public key) cryptosystem based on number theory, which is a block cipher system. It uses two prime numbers to generate the public and private keys. These two different keys are used for encryption and decryption purpose. Sender encrypts the message using Receiver public key and when the message gets transmit to receiver, then receiver can decrypt it using his own private key [12, 13]. RSA operations can be decomposed in three broad steps; key generation, encryption and decryption. RSA have many flaws in its design therefore not preferred for the commercial use. When the small values of p & q are selected for the designing of key then the encryption process becomes too weak and one can be able to decrypt the data by using random probability theory and side channel attacks. On the other hand if large p & q lengths are selected then it consumes more time and the performance gets degraded in comparison with DES. Further, the algorithm also requires of similar lengths for p & q, practically this is very tough conditions to satisfy. Padding techniques are required in such cases increases the system's overheads by taking more processing time.

**Key Generation Procedure:**

1. Choose two distinct large random prime numbers p & q such that $p \neq q$.
2. Compute n= p ×q.
3 .Calculate: phi (n) = (p-1) (q-1).
4. Choose an integer e such that 1<e<phi(n)
5. Compute d to satisfy the congruence relation d × e = 1 mod phi (n); d is kept as private key exponent.
6. The public key is (n, e) and the private key is (n, d). Keep allthe values d, p, q and phi secret.

**Encryption:**

Plaintext: p<n
Ciphertext: C=p^e mod n

**Decryption:**
Ciphertext: C
Plaintext: P=C^d mod n.

**Diffie-Hellman:** This algorithm was introduced in 1976 by Diffie-Hellman. In it, each party generates a key pair and distributes the public key. After obtaining an authentic copy of publickeys, then shared secret can be used as the key for a symmetric cipher.The Diffie-Hellman algorithm grants two users to establish a shared secret key and to communicate over an insecure communication channel . One way authentication is free with this type of algorithm. The biggest limitation of this kind of algorithm is communication made using this algorithm is itself vulnerable to man in the middle attack.

**MD5:** MD5's full form is message-digest algorithm. MD5 is derived from MD4 & was designed by Ron Rivest in 1991 . MD5 is widely used hash function producing a 128-bit hash value, typically expressed in text format as a 32 dig it hexadecimal number. MD5 has been utilized in a wide variety of cryptographic applications, and is also commonly used to verify data integrity.

**KASUMI:** KASUMI[6] is a 64-bit block cipher with a 128-bit secret key. It has a recursive Feistel structure in the same manner as the MISTY1 construction. KASUMI has8 rounds; each round is composed of two functions: the F O function that has 3 rounds of the F I function, and the F L function that has a Feistel structure per forming logical AND/OR operations with subkeys. The order of the two functions depends on the round number: in the even rounds the F O function is applied first, and in the odd rounds the F L function is applied first. The F I function is a 4-round unbalanced Feistel structure using two types of S-boxes, 9 bits and 7 bits in size.

**LUCIFER:**. Lucifer, a direct predecessor of the DES algorithm, is a block-cipher having a 128 bit block size and 128 bit key length. Its general design principles and properties are described and discussed. A simple FORTRAN program is presented which

implements the algorithm, providing a modern, secure cryptographic algorithm that can be used in personal computers. Lucifer is of special interest because it is in the same class of product ciphers as DES but is much simpler. Study of Lucifer may reveal cryptanalytic methods that can be applied to DES.

**CAMELLIA:** We present a new 128-bit block cipher called Camellia[14]. Camellia supports 128-bit block size and 128-, 192-, and 256-bit keys, i.e. the same inter-face specifications as the Advanced Encryption Standard (AES). Efficiency on both software and hardware platforms is a remarkable characteristic of Camellia in addition to its high level of security. It is confirmed that Camellia provides strong security against differential and linear cryptanalysis. Compared to the AES finalists, i.e. MARS, RC6, Rijndael, Serpent, and Twofish, Camellia offers at least comparable encryption speed in software and hardware. An optimized implementation of Camellia in assembly language can encrypt on a Pentium III (800MHz) at the rate of more than 276Mbits per second, which is much faster than the speed of an optimized DES implementation. In addition, a distinguishing feature is its small hardware design. The hardware design, which includes both encryption and decryption, occupies approximately 11K gates, which is the smallest among all existing 128-bit block ciphers as far as we know.

**MARS:** MARS[13] is described as a shared-key (symmetric) block cipher supporting 128-bit blocks and variable key size. MARS is designed to take advantage of the powerful operations supported in today's computers, resulting in a much improved security/performance tradeoff over existing ciphers. As a result, MARS offers better security than triple DES while running significantly faster than single DES. The current C implementation runs at rates of about 65Mbit/sec. on a 200 MHz Pentium-Pro, and 85Mbit/sec. on a 200 MHz PowerPC. In hardware, MARS can achieve a 10*speedup factor. Still, both hardware and software implementations of MARS are remarkably compact, and easily fit on a smartcard and in other limited-resource environments. The combination of high security, high speed, and flexibility, makes MARS an excellent choice for the encryption needs of the information world well into the next century.

**CLEFIA:** CLEFIA[10] is a new 128-bit block cipher supporting key lengths of 128, 192 and 256 bits, which is compatible with AES. CLEFIA achieves enough immunity against known attacks and flexibility for efficient implementation in both hardware and software by adopting several novel and state-of-the-art design techniques. CLEFIA achieves a good performance profile both in hardware and software. In hardware using a 0.09μm CMOS ASIC library, about 1.60Gbps with less than 6Kgates, and in software, about 13 cycles/byte, 1.48Gbps on 2.4 GHz AMD Athlon 64 is achieved. CLEFIA is a highly efficient block cipher, especially in hardware.

**SKIPJACK:** Skipjack[5] is a block cipher that supports a 64-bit block size and a 80-bit key. The block is internally divided into four 16-bit words, where each round applies a keyed non-linear permutation to one word from the block.Skipjack uses two different types of round functions, the A-rounds and the B-rounds. Each encryption consists of a total of 32 rounds, applied in a specific order: First we apply 8 A-rounds, then 8 B-rounds, then another 8 A-rounds, and finally we finish with 8 more B-rounds. We repeat the definitions of the A-rounds and the B-rounds here for convenience.

**GOST 28147-89:** GOST[12] algorithm is a symmetric block cipher, which conforms to Feistel scheme. 64-bit blocks of data are submitted to the input and converted into 64-bit blocks of encrypted data by 256-bit key. In each round the right side of plain text messages is processed by function F, which converts data with three cryptographic operations: adding data and sub key modulo 2^32, substitution of data using S-boxes, and left cyclic shift by 11 positions. Output of F-function is added modulo 2 to the left part of the plaintext, then right and left sides are swapped for next round. The algorithm has 32 rounds. In the last round of encryption right and left parts are not swapped.

GOST uses 8 S-boxes, which convert 4-bit input to 4-bit output. Unlike most encryption algorithms, GOST has no predefined S-boxes and any values can be used for them. Secret key contains 256 bits and is represented as a sequence of eight 32-bit words: K1, K2, K3, K4, K5, K6, K7 and K8. In each round of encryption one of these 32-bit words is used as a round subkey. When round subkey is calculated, the following principle is used: from round 1 to round 24 the order is straight, (K1, K2, K3, K4, K5, K6, K7, K8, K1, K2, etc.); from round 25 to round 32 reversed order is used (K8, K7, K6, K5, K4, K3, K2, K1).

**TURING:** Turing[8] (named after Alan Turing) is a stream cipher designed to simultaneously be:
–Extremely fast in software on commodity PCs,
–Usable in very little RAM on embedded processors, and
–Able to exploit parallelism to enable fast hardware implementation.

**The Turing stream cipher** has a major component, the word-oriented Linear Feedback Shift Register (LFSR), which originated with the design of the SOBER family of ciphers . Analyses of the SOBER family are found in . The efficient LFSR updating method is modelled after that of SNOW .Turing combines the LFSR generator with a keyed mixing-function reminiscent of a block cipher round. The S-box used in this mixing round is partially derived
from the SOBER-t32 S-box . Further aspects of this mixing function have been derived from **Rijndae**l , **Twofish** , **tc24** and **SAFER++.**

Turing is designed to meet the needs of embedded applications that place severe constraints on the amount of processing power, program space and memory available for software encryption algorithms. Since most of the mobile telephones in use incorporate a microprocessor and memory, a software stream cipher that is fast and uses little memory would be ideal for this application. Turing over-comes the inefficiency of binary LFSRs in a manner similar to SOBER and SNOW, and a number of techniques to greatly increase the generation speed of the pseudo-random stream in software on a general processor. Turing allows an implementation tradeoff between small memory use, or very high speed using pre-computed tables. Reference source code showing small memory, key agile,

and speed-optimized implementations is available at , along with a test harness with test vectors. The reference implementation (TuringRef.c) should be viewed as the definitive description of Turing.Turing has four components: key loading, Initialisation vector (IV) loading, an LFSR, and a keyed non-linear filter (NLF). The key loading initializes the keyed S-boxes, and the IV loading initializes the LFSR. The LFSR and NLF then
generate key stream in 160-bit blocks . Five 32-bit words selected from the LFSR are first mixed, then passed through a highly-nonlinear, key-dependent S-box transformation, and mixed again. The resulting 5-wordnonlin-ear blockis combined with 5 new LFSR words to create 160 bits of key stream.The final addition of 5 LFSR words (called whitening) provides the output with good statistical properties, while the nonlinear block hides the linear properties of the LFSR. For each 160-bit block of key stream, the LFSR state is updated 5 times.

**Achterbahn-128/80:** Achterbahn[9] is a stream cipher proposal submitted to the eSTREAM project.After the cryptanalysis of the rst two versions , it has moved on to a newone called Achterbahn-128/80 published in June 2006. Achterbahn-128/80 corresponds to two keystream generators with key sizes of 128 bits and 80 bits,respectively. Their maximal keystream length is limited to 2^63. We present here two attacks against both generators. The attack against the
80-bit variant, Achterbahn-80, has complexity 2^61. The attack against Achterbahn-
128 requires 2^80.58 operations and 2^61 keystream bits. These attacks are based
on an improvement of the attack against Achterbahn version 2 and also on an
algorithm that makes prot of the independence of the constituent registers.

## III. LITERATURE REVIEW

**[1]** Mandal et al. designed an algorithm to merge both RSA algorithm and Diffie-Hellman Algorithm to provide a higher level of data security. Actually, their intent was to secure data of smaller as well as larger size by obtaining one randomly chosen key pair from set of RSA keys and one randomly chosen secret key using Diffie-Hellman algorithm and then applying RSA encryption to make even public components of Diffie-Hellman algorithm inaccessible for any eavesdropper freely.
[2] Wang et al. described a complete set of practical solution to file encryption based on RSA algorithm. With analysis of the present situation of the application of RSA algorithm, they found the feasibility of using it for file encryption..The conventional RSA algorithm used C + + Class Library to develop RSA encryption algorithm and realized Groupware encapsulation with 32-bit windows platform.
[3] Anjula Gupta and Navpreet Kaur Walia have described the various cryptographic algorithms and did a comparative analysis between the different types of cryptography.
[4] Atito et al. proposed a technique which is a composition of both encryption and data hiding using some properties of Deoxyribonucleic Acid (DNA) sequences, consisting of mainly of two phases. In the first phase, the secret data is encrypted using a DNA and Amino Acids-Based Playfair cipher. While in the second phase the encrypted data is steganographically hidden into some reference DNA sequence using an insertion technique.
[5] Knudsen and have examined the structure of the NSA-designed block cipher by cryptanalysing a large number of variants of the algorithm.
[6] Saito in his paper has concentrated on single-key attacks considered to be practical attacks and has proposed a single-key attack on 6-round KASUMI. The attack, which applies a technique of higher order differential attacks, requires 260.8 data and 265.4 encryption time.According to him, the attack is the most powerful single-key attack against reduced-round KASUMI in terms of time complexity.
[7] Zhang and Fu have analysed the PCR code,chaos code, the united chaos encryption algorithm based on logistic map and henon map,etc.
[8] Rose and Hawkes have proposed the very idea of Turing cipher along the analysis of Key Loading and IV loading,Analysis of Whitening and the LFSR(Linear Feedback Shift Registers ) of Turing. Turing offers up to 256-bit key strength, and is designed for extremely efficient software implementation.It combines an LFSR generator based on that of SOBER with a keyed mixing function reminiscent of a block cipher round. Aspects of the block mixer round have been derived from Rijndael, Twofish, tc24 and SAFER++ .
[9] Plasencia has presented two key-recovery attacks against Achterbahn-128/8
[10] Shirai et al. have proposed a new 128-bit blockcipher CLEFIA supporting key lengths of 128, 192 and 256 bits, which is compatible with AES.
[11] Mitali,Kumar and Sharma has analyzed the encryption and decryption time of various algorithms on different settings of data.
[12] Babenko and Maro have done an algebraic analysis of GOST 28147-89 encryption algorithm (also known as simply GOST), which was the basis of most secure information systems in Russia. The general idea of algebraic analysis is based on the representation of initial encryption algorithm as a system of multivariate quadratic equations, which define relations between a secret key and a cipher text. Extended linearization method is evaluated as a method for solving the nonlinear system of equations.
[13] Burwick et al. has described MARS, a shared-key (symmetric) block cipher supporting128-bit blocks and variable key size. MARS is designed to take advantage of the powerful operations supported in today's computers, resulting in a much improved security/performance tradeoff over existing ciphers. As a result, MARS offers better security than triple DES while running significantly faster than single DES. The current C implementation runs at rates of about 65 Mbit/sec. on a 200 MHz Pentium-Pro, and 85 Mbit/sec. on a 200 MHz PowerPC. In hardware, MARS can achieve a 10x speedup factor. Still, both hardware and software implementations of MARS are remarkably compact, and easily fit on a smartcard and in other limited-resource environments. The

combination of high security, high speed, and flexibility, makes MARS an excellent choice for the encryption needs of the information world well into the next century.

[14] Aoki et al. has presented a new 128-bit block cipher called Camellia. Camellia supports 128-bit block size and 128-, 192-, and 256-bit keys, i.e. the same interface specifications as the Advanced Encryption Standard (AES). Efficiency on both software and hardware platforms is a remarkable characteristic of Camellia in addition to its high level of security. It is confirmed that Camellia provides strong security against differential and linear cryptanalysis. Compared to the AES finalists, i.e. MARS, RC6, Rijndael, Serpent, and Twofish, Camellia offers at least comparable encryption speed in software and hardware. An optimized implementation of Camellia in assembly language can encrypt on a Pentium III (800MHz) at the rate of more than 276 Mbits per second, which is much faster than the speed of an optimized DES implementation. In addition, a distinguishing feature is its small hardware design. The hardware design, which includes both encryption and decryption, occupies approximately 11K gates, which is the smallest among all existing 128-bit block ciphers as far as we know.

[15] Amalraj and Jose have given a brief idea bout the different types of Public Key Cryptography.

**IV. COMPARISON TABLE:** The comparison[3] is shown in Table 1

| Algorithm | Created By | Year | Key Size(bits) | Block (bits) | Round | Structure | Flexible | Features |
|---|---|---|---|---|---|---|---|---|
| KASUMI | SAGE | - | 128 | 64 | 8 | Feistel | No | MISTY better |
| Clefia | SONY | 2007 | 128/192/256 | 128 | 18/22/26 | Feistel | Yes | CRYPTEC uses |
| Skipjack | NSA | 1998 | 80 | 64 | 32 | Unbalanced Feistel | No | NIST recoomended not to use skipjack |
| Turing | Qualcomm Australia | 2003 | 256 | - | - | Fast Stream Cipher | Yes | 5.5 cycles/byte speed |
| Achterbahn-128/80 | eCRYPT | 2006 | 80/128 | - | - | Synchronous Stream Cipher | - | No known crptanalytic attacks |
| GOST-28147-89 | USSR,KGB | 1994 | 256 | 64 | 32 | Feistel | No | Deeply flawed |
| Lucifer | Horst Feistel | 1971 | 48/64/128 | 48/32/128 | 16 | Synchronous Stream Cipher,Feistel | Yes | Succeded by DES |
| Mars | IBM | 1998 | 128/192/256 | 128 | 32 | Type3 Feistel | Yes | Meet-in-the-middle breaks 21 rounds |
| Camellia | Mitsubishi Electric | 2000 | 128/192/256 | 128 | 18/24 | Fesitel | Yes | Theoretically breakable by EXTENDED SPARSE LINEARISATION |
| DES | IBM | 1975 | 64 bits | 64 bits | 16 | Feistel | No | Not Strong Enough |
| 3DES | IBM | 1978 | 112 or 168 | 64 bits | 48 | Feistel | Yes | Adequate Security |
| AES | Joan Daemen & incent Rijmen | 1998 | 128, 192, 256 bits | 128 bits | 10,12, 14 | Substitution-Permution | Yes | Replacement for DES, Excellent Security |
| Blowfish | Bruce Schneier | 1993 | 32-448 | 64 bits | 16 | Feistel | Yes | Excellent Security |
| RC4 | Ron Rivest | 1987 | Variable | 40-2048 | 256 | Feistel Stream | Yes | Fast Cipher in SSL |
| RC2 | Ron Rivest | 1987 | 8-128 64 by default | 64 bits | 16 | Feistel | - | Stream Cipher |
| Twofish | Bruce Schneier | 1993 | 128- 256 | 128 bits | 16 | Feistel | Yes | Good Security |
| Serpent | Anderson,, Lars Knudsen | 1998 | 128- 256 | 128 bits | 32 | Substitution Permutation | Yes | Good Security |
| IDEA | James Massey | 1991 | 128 bits | 64 bits | 8.5 | Substitution Permutation | No | Not Strong Enough |
| RC6 | Ron Rivest, Matt Robshaw | 1998 | 128 bits to 256 bits | 128 bits | 20 | Feistel | Yes | Good Security |
| RSA | Rivest,, Shamir, Adleman | 1977 | 1,024 to 4,096 | 128 bits | 1 | Public Key algorithm | No | Excellent Security, low speed |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Diffie Hellman | Whitfield Diffie , Hellman | 1976 | 1024 to 4096 bits | 512 | - | Asymmetric algorithm | Yes | Many attacks |
| MD5 | Ronald Rivest | 1992 | Series of MD | 512 | 4 | Merkle–Damgård construction | | Hash Function |

## V. SCOPE – WHAT LIES AHEAD

We have covered a variety of Cryptographic techniques along with their advantages as well as drawbacks. Though the classical methods are special in their own rights due to historical reasons as well as the amount of secrecy they provide, it can not be denied that the modern cryptographic landscape is almost totally driven by computers. If we are to be brave enough to move a bit away from the DES,RSA,AES and Blowfish and it's successor ,Twofish,we'll find that something called **Quantum Cryptography** is trying to make it's way-and for good reasons too. All the current public key encryption as well as signature schemes like RSA and ElGamal which is based on the Diffiie-Hellman can all be pretty easily broken by their quantum counterparts. It is impossible to copy the data encoded in the Quantum state and even if someone tries to read it,the state will change. The most widely used is the quantum key distribution which offers an information-theoretically secure solution to the key exchange problem. Now there are two keywords we would like to focus on.The first one is information-theoretically secure solution .Invented by Shannon as a part of the Information Theory,it was originally used to prove the unbreakability of the one time pad. Basically, an information theoretically secure cryptosystem is unbreakable by an cryptanalytic method irrespective of the amount of quantum computing power the eavesdropper might have. It is quite an integral part of Shamir's secret sharing scheme .At the heart of quantum cryptography lies the advantage of qubits and the fact that it is impossible to copy qubits. Using the Bloch sphere representation in the Hopf co-ordinates it can be shown that qubits are quite like classical bits which can only be 0 or 1.However,qubits are a bit different in the sense that it supports superposition of both 0 and 1.A qubit can hold more information is superdense coding is used. Multiple qubits also support quantum entanglement. This is instrumental in achieving quantum analysis which uses Ancilla qubits. Quantum coin flipping also uses the idea of qubits to transfer information between two parties who do not trust each other. The main application of quantum cryptography lies in the Quantum Key Distribution(QKD) which has "unconditional security" and it is secure against quantum computers too.**BB84** is an example of the QKD which was the very first Quantum Cryptography protocol. It was invented by Charles Bennett and Gilles Brassard in 1984.Now,there are different types of Quantum Cryptography like Position absed,Device Independent and the Post-quantum Cryptography.All of them are equally important and require much more research in their respective fields.

Another new and upcoming type of cryptography that demands our attention is **Chaotic Cryptology**.Though the cryptosystems based on the mathematical chaos theory are relatively unsecure,this type of cryptography holds massive potential.The only hindering element to it's acceptance is the fact that it is,at the moment very difficult to implement chaos maps that generate entropy that matches the required **Confusion and Diffusion.** In Shannon's original definitions, *confusion* refers to making the relationship between the ciphertext and the symmetric key as complex and involved as possible; *diffusion* refers to dissipating the statistical structure of plaintext over the bulk of ciphertext. This complexity is generally implemented through a well-defined and repeatable series of *substitutions* and *permutations* If chaotic parameters, as well as cryptographic keys, can be mapped symmetrically or mapped to produce acceptable and functional outputs, it will be very difficult for an adversary to find the outputs if he doesn't know the individual values.There are different types of Chaotic Cryptography such as Chaos-Based-image Encryption,Chaos-Based-Hash Functions and Chaos-Based-Random-Number Generation.Implementing chaotic maps correctly is a struggle but nonetheless,considerable amount of work is being done on it.

Now,we would like to turn our focus onto something called **Elliptic-Curve Cryptography or ECC**.They are a type of public key cryptography based on algebraic structures of elliptic curves on finite fields.They have smaller keys compared to non ECC systems which are based on palne Galois fields.A Glaois field is one which has a finite number of elements. Elliptic curves are applicable for key agreement, digital signatures, pseudo-random generators,etc. Indirectly, they can be used for encryption by combining the key agreement with a symmetric encryption scheme. They are also used in several integer factorization algorithms based on elliptic curves that have applications in cryptography, such as Lenstra elliptic-curve factorization or the **elliptic-curve factorization method** (**ECM**).ECM is the third fastest general purpose algorithm after multiple polynomial quadratic sieve and the genral number field sieve.The most used curves are the **Edwards curves** and the **Montegomery** or the **Weierstrass** curves.Recent developments use the hyperelliptic curves(HECM-Hyper Elliptic Curve method).There is a quantum version of ECM with Edwards curves using the Grover's algorithm invented by **Bernstein, Heninger, Lou, & Valenta.** For elliptic-curve-based protocols, it is assumed that finding the discrete logarithm of a random elliptic curve element with respect to a publicly known base point is infeasible: this is the "elliptic curve discrete logarithm problem" (ECDLP). The security of elliptic curve cryptography depends on the ability to compute a point multiplication and the inability to compute the multiplicand given the original and product points. The size of the elliptic curve determines the difficulty of the problem. Some discrete logarithm-based protocols are:

- The Elliptic Curve Diffie Hellman (ECDH) key agreement scheme is based on the DH scheme,
- The Elliptic Curve Integrated Encryption Scheme (ECIES), also known as Elliptic Curve Augmented Encryption Scheme or simply the Elliptic Curve Encryption Scheme,
- The Elliptic Curve Digital Signature Algorithm (ECDSA) is based on the Digital Signature Algorithm,
- The deformation scheme using Harrison's p-adic Manhattan metric,
- The Edwards-curve Digital Signature Algorithm (EdDSA) is based on Schnorr signature and uses twisted Edwards curves,
- The ECMQV key agreement scheme is based on the MQV key agreement scheme,

- The ECQV implicit certificate scheme.

In cryptography, **Curve25519** is an elliptic curve offering 128 bits of security and designed for use with the elliptic curve Diffie–Hellman (ECDH) key agreement scheme. It is one of the fastest ECC curves and is not covered by any known patents.

. There's another emerging trend in Cryptography and that is **DNA Cryptography**[4][7].Deoxyribonucleic acid (DNA) is the master molecule whose structure encodes all the information needed to create and direct the chemical machinery of life. In 1953, the structure of DNA was correctly predicted by Watson and Francis Crick that DNA molecule consists of two long polynucleotide chains each of these chains is known as a DNA chain, or a DNA strand which is made from simple subunits, called nucleotides. Each nucleotide consists of a sugar-phosphate molecule with a nitrogen-containing side group, or base. The bases are of four types (adenine, guanine, cytosine, and thymine), corresponding to four distinct nucleotides, labeled A, G, C, and T.

The DNA-based cryptography is a new and very promising direction in cryptography research. DNA can be used in cryptography for storing and transmitting the information, as well as for computation. Recently, a number of cryptographic techniques have been proposed to utilize the DNA digital format in the ciphering process itself.

Although different methods of Data hiding techniques were introduced including: invisible inks, microdots, digital signatures, and spread spectrum communications, DNA-based Data hiding techniques have been recently added to that list. These techniques depend on the high randomness of the DNA to hide any message without being noticed. In fact, DNA has many characteristics which make it a perfect Data hiding media. These characteristics have two significant facts; the DNA has tremendous information storage capacity. In addition, any DNA sequence can be synthesized in any desirable length.Inspired by the microdots used during the 2nd world war, Clelland et al. developed an extension of it using DNA. Leier et al. encoded binary information into DNA sequences. The resulting DNA sequence is mixed with dummy strands and can only be detected and isolated if the primer sequence is known.Saeb et al. proposed two Biotechnological methods for hiding message into DNA using DNA Recombinant and DNA Mutagenesis.

## VI. CONCLUSION

Cryptographic landscape is witnessing advancements in a way like never before. Newer and advance methods are cropping up more frequently.In this paper,we have ventured to present Cryptography with all its facets.In this study of reverie,we have covered aspects of classical cryptographic approach as well the modern ones.The possibilities in this field are endless.Security and privacy is not for granted,especially in today's world where information is power.Cryptography along with crypatanalysis are very useful tools that make a huge difference and as such should be used with the utmost care and caution.

## VII. ACKNOWLEDGMENT

We would like to take this opportunity to thank our professors and friends,all of whom share the same passion for Cryptography and without whose help,it would have been quite impossible to write this paper.

## VIII. REFERENCES

**[1]** Mandal, B.K. , Bhattacharyya , Bandyopadhyay S.K. , "Designing and Performance Analysis of a Proposed Symmetric Cryptography Algorithm ", Communication Systems and Network Technologies (CSNT), 2013, pp. 453 – 461.

[2] Wang, Suli , Liu, Ganlai , "File encryption and decryption system based on RSA algorithm", Computational and Information Sciences (ICCIS), 2011, pp. 797 – 800.

[3] Anjula Gupta,Navpreet Kaur Walia ," Cryptography Algorithms: A Review", 2014 IJEDR,Volume 2, Issue 2,ISSN: 2321-9939

[4] A. Atito,A. Khalifa,S. Z. Rida," DNA-Based Data Encryption and Hiding Using Playfair and Insertion Techniques", J. of Commun. & Comput. Eng ISSN 2090-6234 www.m-sciences.com Volume 2, Issue 3, 2012, Pages 44: 49.

[5] Lars Knudsen , David Wagner, "On the structure of Skipjack", Discrete Applied Mathematics 111 (2001) 103–116.

[6] Teruo Saito, "A Single-Key Attack on 6-Round KASUMI",IACR.

[7] Yunpeng Zhang and Liu He Bochen Fu, "Research on DNA Cryptography", Applied Cryptography and Network Security.

[8] Gregory G. Rose and Philip Hawkes, "Turing: a Fast Stream Cipher", Qualcomm Australia.

[9] María Naya-Plasencia, "Cryptanalysis of Achterbahn-128/80", INRIA, projet CODES, Domaine de Voluceau.

[10] Taizo Shirai1, Kyoji Shibutani, Toru Akishita, Shiho Moriai, and Tetsu Iwata." The 128-bit Blockcipher CLEFIA (Extended Abstract)",SONY Corporation.

[11] Mitali, Vijay Kumar and Arvind Sharma," A Survey on Various Cryptography Techniques", International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), Volume 3, Issue 4, July-August 2014,ISSN 2278-6856

[12] Ludmila Babenko, Ekaterina Maro," Algebraic Cryptanalysis of GOST Encryption Algorithm", Journal of Computer and Communications, 2014, 2, 10-17 Published Online March 2014 in SciRes.

[13] Carolynn Burwick,Don Coppersmith,Edward D'Avignon, Rosario Gennaro, Shai Halevi, Charanjit Jutla ,Stephen M. Matyas Jr. ,Luke O'Connor ,Mohammad Peyravian ,David Safford and Nevenko Zunic," MARS - a candidate cipher for AES",IBM Corporation.

[14] Kazumaro Aoki, Tetsuya Ichikawa,Masayuki Kanda,Mitsuru Matsui,Shiho Moriai, Junko Nakajima and Toshio Tokita," Camellia: A 128-Bit Block Cipher Suitable for Multiple Platforms — Design and Analysis —"

[15] A. Joseph Amalraj, Dr. J. John Raybin Jose," A SURVEY PAPER ON CRYPTOGRAPHY TECHNIQUES", International Journal of Computer Science and Mobile Computing, Vol.5 Issue.8, August- 2016, pg. 55-59