Comparative Analysis of Block Key Encryption Algorithms

Mukta Sharma
Research Scholar
Teerthanker Mahaveer
University, Moradabad

R. B. Garg, PhD
Ex-Professor
University of Delhi
Delhi

ABSTRACT

Humankind has grown phenomenally from Stone Age to information technology age. Information technology is deeply treasured; mainly because of the Internet. The internet has brought great changes in almost every domain of life. Besides a computer, laptop, Desktop, even a Mobile Phones, I-Pad, IPod have the facility to use the Internet. With the Internet, life has become easier to get things done at a great speed and effortlessly. Such as searching for some information, retrieving maps, images, bookings for air tickets, movie tickets, sports or concerts tickets, hotel reservations, grocery, vegetables, and fruits, paying bills online, transferring money without going to the bank, etc. and the list is endless. It is a well-known phrase “Each coin has two sides”; similarly, with the benefits of using the internet, there are many threats or flaws. The Internet has given easy access to attack our personal information, financial transactions, etc. by hackers, pedophiles, net extortion, and salami attacks and so on. There are numerous attacks, and the researchers are trying to figure out some solution for them. One of the most popular ways to ensure security is using cryptography.

This research paper is written with intent to explore symmetric key encryption algorithms. The research paper will focus on numerous symmetric key algorithms which have been ensuring security while transacting funds online. A study on block cipher algorithms gave a horizon to do a comparative analysis based on its working, real-time implementation, architecture, performance/scalability, flaws, etc.

General Terms

Block Ciphers, Cryptography, Symmetric Key Algorithm, etc.

Keywords

Advanced encryption Standard (AES), Asymmetric Key, Blowfish, Cryptography, Data Encryption Standard (DES), IDEA, Symmetric Key, Triple Data Encryption Standard (3DES)

1. INTRODUCTION

According to NIST, Computer Security Handbook [NIST95] defines the term computer security as “The protection afforded to an automated information system in order to attain the applicable objectives of preserving the integrity, availability, and confidentiality of information system resources (includes hardware, software, firmware, information / data, and telecommunications)”[4].Security is one of the most significant topics of research and discussions. Following objectives have been briefly explained regarding requirements and the definition of a loss of security:-

• Confidentiality: focus on two important traits one is data confidentiality, and the other is Privacy. Data confidentiality assures that information is confidential and is accessed by authorized users only. Whereas, Privacy ensures individuals to control the rights of collecting, storing, and disclosing their personal information to only those whom they wish to share [9] [26].

In short, confidentiality can be defined as preserving authorized restrictions on accessing and revealing the information (it may be personal or proprietary information). A loss of confidentiality is the unauthorized disclosure of information.

• Integrity: highlights two traits data integrity and system integrity. Data integrity: Promises that information and programs are consistent, trustworthy and reliable. System integrity: Should guarantee that a system performs is working in a proper manner, which means it should not allow unauthorized manipulation of the system [9] [26].

In Brief, Integrity is protecting against improper information modification or destruction and inaccessibility by unauthorized users. Loss of integrity allows any unauthorized modification or destruction of information [26].

• Availability is the assurance of giving the desired, correct information timely to the authentic user.

In short, Availability is an agreement of accessing accurate information to the authorized user. A loss of availability is the disruption of access to information or an information system [9] [26].

Cryptology is the study of reading, writing and breaking of codes. It comprises of cryptography (secret writing) and cryptanalysis (breaking code) as shown in figure 1. Symmetric cryptographic algorithms; which are used to hide messages have been the focus of research. Cryptography has given a platform which can ensure not only confidentiality but also integrity, availability, and non-repudiation of messages/ information.

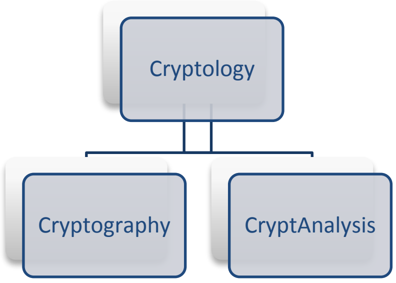


Fig. 1: Classification of Cryptology

Two Greek words Crypt (Secret) and graphics (Writing) makes Cryptography, means secret writing. It is one of the most important techniques to deal with security, especially in transacting information online [22]. Encryption and Decryption are two processes used to perform a cryptographic algorithm. Encryption will scramble the plain text or message or information; into Ciphertext (scrambled text). Decryption is the reverse process used to retrieve the plain text from the cipher (scrambled text) [26]. Symmetric Key Encryption Algorithm, Asymmetric Key Encryption Algorithms, and Protocols are the way of implementing Cryptography as shown in figure 2. Symmetric Key (the same key is shared for communication by both sender and receiver to encipher and decipher the text). Unlike Symmetric Key, Asymmetric Key uses a set of keys (one public key-globally announced and another private key used to decrypt the message). Stream Cipher and Block Ciphers are two separate ways of classifying Symmetric Key. Stream Cipher as the name specifies a single stream of data or information is encrypted and then decrypted at a time. Block Cipher works with block or chunks of data or message instead of a single stream, character, or byte. This paper has focused primarily on block ciphers [24]. Block ciphers can build strong or secure algorithm by using Claude Shannon two atomic operations multiple times, two services are:-

- Confusion- Is mapping of one value to another, Obscuring (hiding) relationship between Plain Text and Cipher Text. Substitution is a good example of Confusion (Substitution Table, Lookup Table, S-Boxes, etc.) [24].
- Diffusion- The influence of one each plain text bit is spread over many cipher text. Reordering of bit position for each of the inputs. Example- Permutation

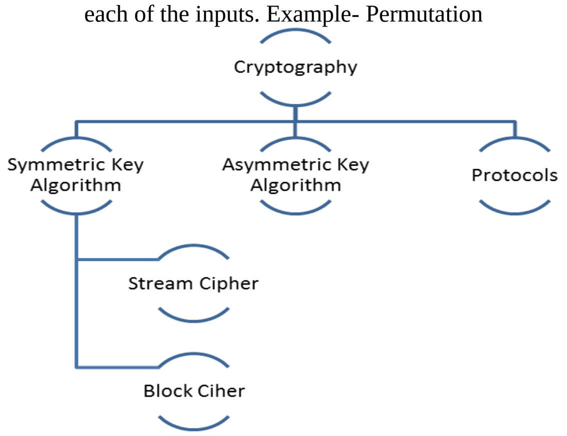


Fig. 2: Organization of Cryptography

2. CLASSIFICATION OF BLOCK CIPHER ALGORITHMS

In the market there are various block cipher algorithms, out of which some are patented, some algorithms were used extensively earlier, and few are widely acceptable even now. This paper will shed light on the following algorithms depicted in figure 3.

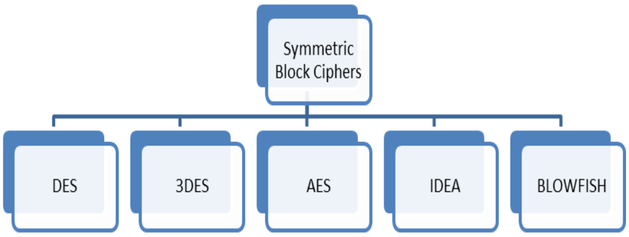


Fig. 3: Symmetric Key Block Ciphers

3. COMPARATIVE ANALYSIS

For ensuring security Cryptography is considered to be apt especially Symmetric Key Block Cipher algorithms are highly recommended. The paper focuses on the key aspects, factors, and varied parameters for analyzing the algorithms. They can be categorized as depicted in figure 4 [3] [5] [18] [19].

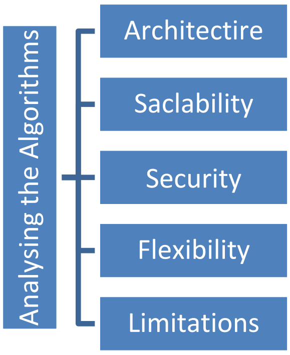


Fig. 4: Testing Parameters for Algorithms

1. **Architecture** –The basic structure involves its characteristics like whether it is a symmetric key algorithm or asymmetric key algorithm. The architecture deals with the specifications like key size, network type (Feistel network or substitution-permutation network), functions or operations it can perform and how they are implemented, how many rounds, etc. In short, architecture means the basic structure and layout of the algorithm [3] [5] [18] [19].
2. **Scalability** – It is one of the key element on which encryption algorithms can be analyzed. It is used to analyze the performance of the algorithm based on certain parameters such as Memory Consumption, Encryption rate, Software-Hardware performance; Computational efficiency, Mode, Data File type, speed, throughput, etc. [3] [5] [18] [19].
3. **Security-** It is one of the most significant parts of a cryptography algorithm which resists an attack. Key Length plays a pivotal role in context to Security. The greater the key length, difficult it becomes to crack. If an algorithm use Confusion and Diffusion it becomes little safer. A number of rounds enhance the security, if an algorithm uses Confusion, Diffusion multiple times, it can ensure better security, therefore, it is said to be more secure with number of rounds. But an increasing number of rounds makes the process very slow [22] [24].
4. **Flexibility-**This determines whether the algorithm can endure minor modifications according to the requirements
5. **Limitations (Known Attacks)** – This defines how excellent the algorithm works by using the computer resources available to it. Further, how often it is vulnerable to different types of attacks.

4. WORKING AND IMPLEMENTATION

1) **DES** stands for Data Encryption Standard designed by IBM in 1972 and was later adopted as a standard by US government in 1974 published by NIST [7] [17] [21] [29].

DES works on Feistel network. 64 bit Plain Text is given as an input and 64 bit Cipher Text comes as an output. Initially, Key is of 64 bits; parity bits (8, 16, 24, 32, 40, 48, 56, and 64) are removed. Later computation is carried on 56 bits key. PC-1 is used just once and PC-2 is used every time to compute a new sub key of **48 bits, instead of 56 bits**. 16 rounds are executed, Left Circular Shift is also based on the rounds like 1, 2, 9 and 16 move 1left bit and rest all rounds (3, 4, 5, 7, 8, 10, 11, 12, 13, 14 and 15) move 2 bits which finally makes 28bit (4*1+ 12*2= 28 bits)

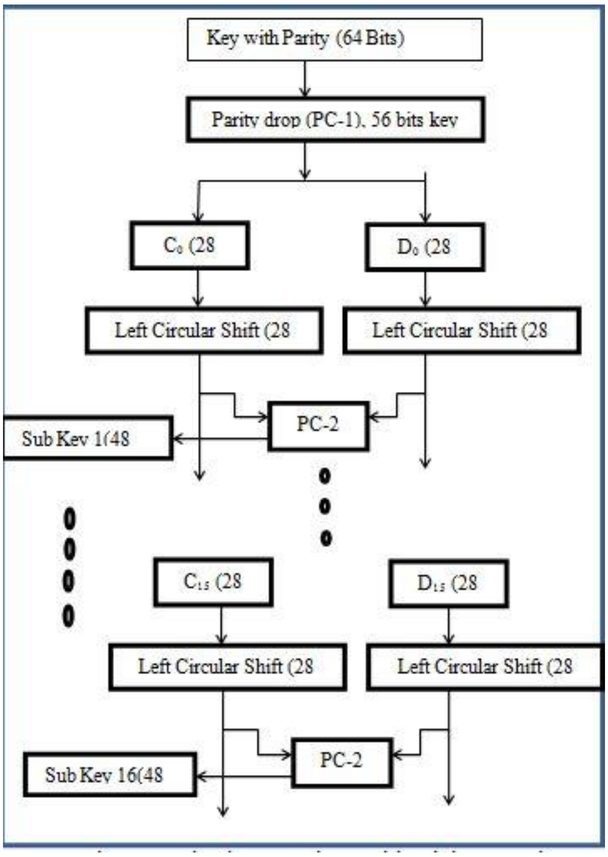


Fig. 5 Sub Key Generation

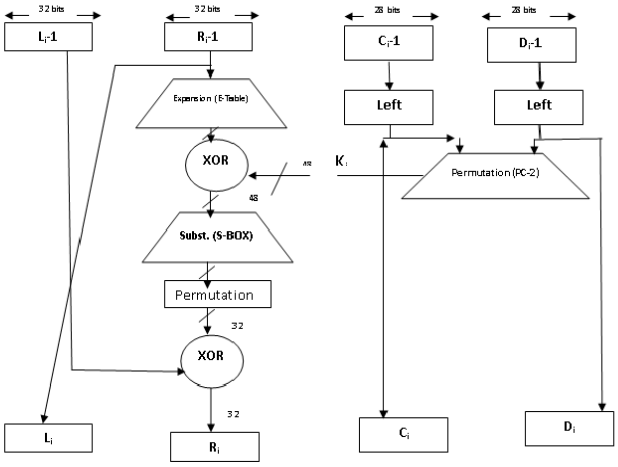


Fig. 6 DES Rounds

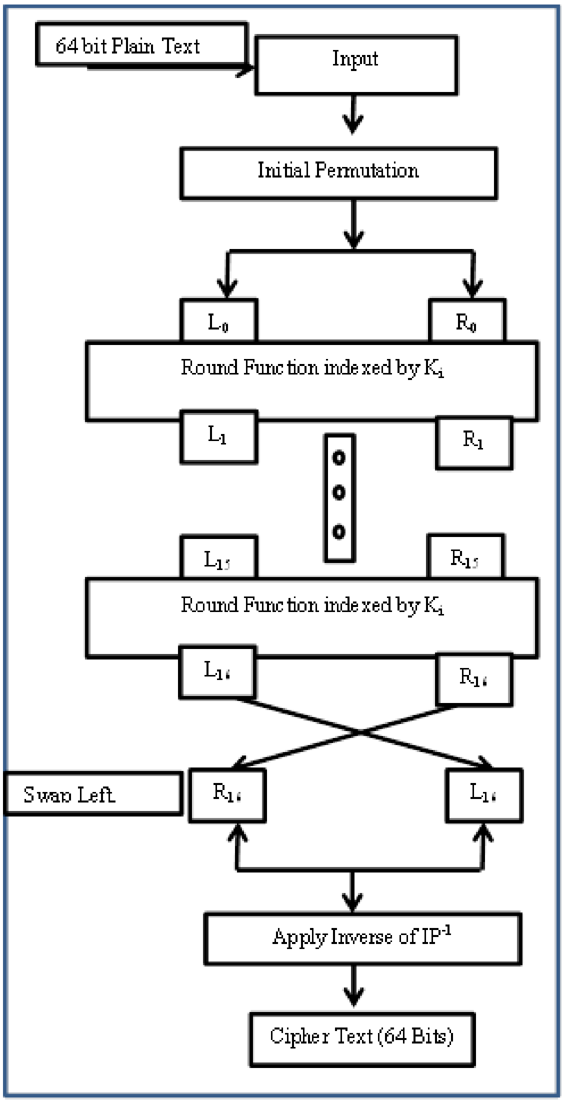


Fig. 7 DES Working

- **Rounds-** Initially, R is 32 bits, and it needs to be expanded to 48 bits so that XOR can be performed. Initially, the R is 4*8=32, and Expansion Matrix needs to expand it to 6*8=48. For the expansion, one previous bit is concatenated as the first bit, and one next bit is added as the last bit like 1234 will be expanded as 32(previous bit)1234 and 5(next bit)

Table 1: Expansion Permutation Table

1	2	3	4	5	6	7	8	Remark
1234							29303132	4*8=32
3212345							**28**29303132 **1**	6*8=64

- XOR is also known as Modulo 2.
- S-Box substitutes 48-bit input to 32bit output. Eight blocks of 6 bits (i.e. 48 bits). Each S-Box (Total 8 S-Boxes) will generate a 4-bit output (4*8=32). First and last bit of a block are used for the row. Middle 4 values of a block are used to represent

column and then using S-Box table can take the value and convert it to a hexadecimal value.

Actual working of DES

- 64 bits are given as an input
- Initial permutation is carried on 64 bit block.
- Split into two halves of 32 bits each (L_0,R_0)
- R (L_i, R_i) passed to 16 rounds. Rounds are identical.
- After last round L_{16} & R_{16} are swapped to create pre output
- Inverse permutation is performed. Plain Text is finally converted to Cipher Text.

2) **Need for any other Algorithm**- After more than 20 years of successful journey of DES. The exhaustive key search started causing its discomfort. In Dec 1988 Deep Crack; special purpose DES hardware cracker was made for $2, 50, 000. In 1990 two Israeli researchers with the help of Differential Cryptanalysis were able to break using Known Plain Text attack. In the year 1993, one Japanese scientist worked on Linear Cryptanalysis. Later in the year 2007 software named COPACOBAMA was introduced for just $70,000 to break DES. **Triple DES or 3DES**- In November, 1998 to ensure security 3DES was introduced as an enhancement of Data Encryption Standard. It was not feasible regarding money & popularity and usage of DES to completely abandon DES. DES was widely adopted by large security architectures. Therefore, the manner was changed instead of running DES just once; it was decided to run it three times for better security. It led to the modified schemes of Triple DES (sometimes known as 3DES) [29].

This method is similar to the original DES, but it is applied three times (performs three iterations) to increase the security. 3DES uses 64-bit block size as an input and give 64 bit as an output, 16*3=48 rounds; and a key length of 168 bits (56*3). The 3DES encryption algorithm works in a sequence Encrypt-Decrypt-Encrypt (EDE). The decryption process is just reverse of Encryption process (Decrypt- Encrypt-Decrypt). 3DES is more complicated and designed to protect data against different attacks. 3DES has the advantage of the reliability and a longer key length that eliminates many attacks like brute force. 3DES higher security was approved by the U.S. Government. Triple DES has one significant limitation; it is much slower than other block encryption methods.

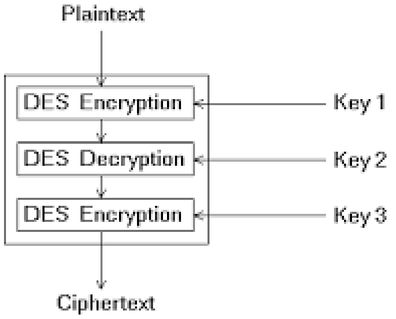


Fig. 8 3DES

- **The encryption algorithm is:**

Ciphertext = EK3(DK2(EK1(Plaintext)))

As the display in figure 8, the plain text is encrypted with DES along with Key, K1 then the ciphertext of the first round will be decrypted with Key, K2 and lastly the result of round 2; the plain text will be encrypted with Key, K3 to get the final Ciphertext.

- **The decryption algorithm is:**

Plaintext = DK1(EK2(DK3(Ciphertext)))

Decryption works in a reverse direction. In this, the Ciphertext decrypt with key, K3 to generate a plain text which will then be encrypted with Key, K2 to get cipher text and then finally cipher text will be decrypted with Key, K1 to get the final plain text.

The standards define three keying options:

- Keying option 1-All three keys are different. Keying option 1 is the strongest, with 3 × 56 = 168 independent key bits. Secure but time-consuming.
- Keying option 2- K1 and K2 are independent, and K3 = K1. It is less secure with 2*56=112 independent key bits.
- Keying Option 3- All three keys are identical, i.e. K1 = K2 = K3. It is equivalent to DES, with only 56 bits. Not recommended by NIST.

3) **IDEA**- (International Data Encryption Algorithm) is a symmetric key block cipher algorithm, developed at ETH in Zurich, Switzerland. IDEA was designed by Xuejia Lai and James Massey. It was published in 1991 and was initially named as Improved Proposed Encryption Standard (IPES) as it was the modified version of Proposed Encryption Algorithm. The name was changed in 1992 to International Data Encryption Algorithm. It is based on substitution-permutation structure. It is a block cipher that uses a 64-bit plain text, divided equally into 16bits each (16*4=64); with eight rounds and a Key Length of 128-bits. For each round 6 sub keys are required four before the round and two within the round (8*6= 48 sub keys+ 4 sub keys are used after the last or eighth round that makes total 52 sub- keys). IDEA does not use S-boxes. IDEA uses the same algorithm in reverse order for decryption [1] [28].

- **Keys**

A 128-bit key is split into 16 bits sub-keys from K1 to K8. Then the bits are shifted to the left 25 bits, ensuring that repetition does not occur in the sub keys. The resulting 128-bit string is split into eight 16-bit blocks that become the next eight sub keys. The shifting and splitting process is repeated until 52 sub keys are generated.

- **Actual Working of IDEA**
  - The 64 bit plaintext is divided into four blocks (A, B, C, and D)
  - 128 bit key is divided into 52 sub-keys (K1..K52)
  - 8 rounds with 6 sub keys= 4 keys before the round and 2 keys in the round
- Round 1 (there are 8 round and they are identical)
  - Multiply A by K1, Add K2 to B, Add K3 to C, Multiply D by K4.
  - Compute E=A XOR C and F= B XOR D.

- Multiply E by K5. Add the new value of E to F.
- Now multiply F by K6. Add the result of F, to E.
- Then XOR F with A and with C and XOR E with B and D separately.
- Swap B and C.
- Repeat this entire seven more times, using K7 through K48. After the eighth round do not swap B and C.
- Last step would be multiplying A by K49. Add K50 to B. Add K51 to C. Multiply D by K52.

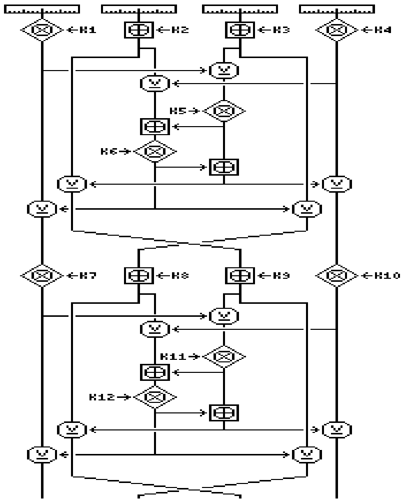
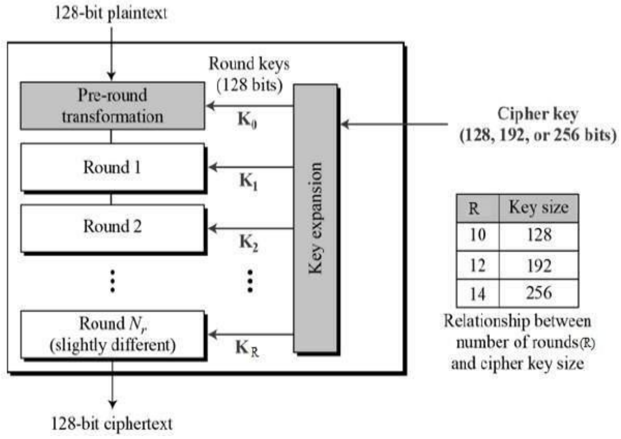


Fig. 9: IDEA

4) **AES-** In the year 1998, NIST conducted a competition open for all, where they accepted many algorithms, out of which they selected 15 algorithms. After analysis by the cryptographic research community five best cryptography algorithms (MARS, RC6, Rijndael, Serpent, and Twofish) were shortlisted. In 2000, NIST declared Rijndael as a new Advanced Encryption Standard and on May 26, 2002, it became effective as a federal government standard. Rijndael was proposed by Joan Daemen and Vincent Rijmen. AES is also a symmetric key algorithm based on the substitution–permutation Network [2] [8] [12] [32].



R	Key size
10	128
12	192
14	256

Relationship between number of rounds(R) and cipher key size

Fig. 10 AES

As depicted in Fig. 10, AES use a 128-bit block as plain text, which is organized as 4*4 bytes array also called as State and is processed in several rounds. It has variable Key length 128, 192 or 256-bit keys. Rounds are variable 10, 12, or 14 depends on the key length (Default # of Rounds = key length/32 + 6). For 128 bit key, a number of rounds are 10; 192 bit key, 12 rounds and for 256 bit key, 14 rounds. It only contains a single S- box (which takes 8 bits input, and give 8 bits output) which consecutively work 16 times. Originally the cipher text block was also variable, but later it was fixed to 128 bits.

The Encryption and decryption process consists of 4 different transformations applied consecutively over the block data bits, in a fixed number of iterations, called rounds. The four different changes are described in detail below:

- Sub Bytes function performs a non-linear substitution of bytes that operates independently on each byte of the State using a substitution table (S-box). 4*4 matrixes are used for substitution. There are 16 parallel S-boxes each with eight inputs and eight outputs. S-boxes are based on two mathematical properties Multiplication and Affine Transformation. This S-box which is invertible is constructed by first taking the multiplicative inverse in the finite field GF (28) with irreducible polynomial m(x) = x8 + x4+ x3 + x + 1. The element {00} is mapped to itself. Then affine transformation is applied (over GF (2)).
- Shift Rows function performs byte-wise circular shifts on last three rows of the state. In this function, first row is not shifted, the second row is shifted 1 byte left the position, the third row is shifted left by two positions and in the fourth row, 3 bytes are shifted leftward.
- Mix Columns function operates on each State column-by-column, treating each column as a four-term polynomial. The columns are considered as polynomials over GF (28) and multiplied by modulo x4 + 1 with a fixed polynomial a(x) = {03} x3+ {01} x2+ {02} x.
- Add Round Key is the final function; in this, a 16 byte Key is simply XOR with the 16-byte state. Each Key is obtained from Key expansion algorithm.
- Key Expansion: The purpose is to calculate round key for each round, based on the original key. This function comprises of three sub-functions:-
  a. Sub Word takes a four-byte input and applies S-box to each of four bytes to produce an output word.
  b. Rot Word takes a word, performs cyclic permutation, and returns the word.
  c. Round Const function contains a round constant array that performs a bitwise XOR function. Round constant array contains values given by ½xi_1; f00g; f00g; f00g_ with xi_1 being powers of x (x is denoted as {02}) in the field GF (28).

The decryption process is the direct inverse of the encryption process. Hence the last round values of both the data and key are first round inputs for the decryption process and follows in decreasing order.

AES is extremely fast and compact cipher. For implementers its symmetric and parallel structure provides excellent and an active resistance against cryptanalytic attacks. The larger block size prevents birthday attacks, and large key size prevents brute force attacks

5) BlowFish

Blowfish is a symmetric block cipher, designed in 1993 by Bruce Schneier. It works on of 64-bit block size. Key length is variable from 32 bits to 448 bits. It has16 rounds and is based on Feistel network. It has a simple structure, and it's easy to implement. It encrypts data on 32-bit microprocessors at a rate of 18 clock cycles per byte so much faster than AES, DES, and IDEA. Since the key size is large, it is complex to break the code in the blowfish algorithm. It is vulnerable to all the attacks except the weak key class attack. It is unpatented and royalty-free. It requires less than 5K of memory to run Blowfish [23] [30].

- **Sub-Key**

Blowfish uses a large number of sub-keys. Subkeys must be pre-computed before any data encryption or decryption. The sub-keys are calculated using the Blowfish algorithm:

1. Initialize first the P-array and then the four S-boxes, in order, with a fixed string. This string consists of the hexadecimal digits of pi (less than initial 3): P1 = 0x243f6a88, P2 = 0x85a308d3, P3 = 0x13198a2e, P4 = 0x03707344, etc.
2. XOR P1 with the first 32 bits of the key, XOR P2 with the second 32-bits of the key, and so on for all bits of the key (possibly up to P14). Repeatedly cycle through the key bits until the entire P-array has been XOR with key bits. (For every short key, there is at least one equivalent longer key; for example, if A is a 64-bit key, then AA, AAA, etc., are equivalent keys.)
3. Encrypt the all-zero string with the Blowfish algorithm, using the sub-keys described in steps (1) and (2).
4. Replace P1 and P2 with the output of step (3).
5. Encrypt the output of step (3) using the Blowfish algorithm with the modified sub keys.
6. Replace P3 and P4 with the output of step (5).

Continue the process, replacing all entries of the P array, and then all four S-boxes in order, with the output of the continuously changing Blowfish algorithm.

In total, 521 iterations are required to generate all required sub-keys. Applications can store the sub keys rather than execute this derivation process multiple times.

- **Data encryption:**
  - The input is a 64-bit data element, X.
  - Split X into two halves of 32-bits each: XL, XR.
  - Then, for i = 1 to 16
    - XL = XL XOR Pi
    - XR = F(XL) XOR XR
    - Swap XL and XR
  - After the sixteenth round, swap XL and XR again to undo the last swap.
  - Then, XR = XR XOR P17 and XL = XL XOR P18.
  - Finally, recombine XL and XR to get the Ciphertext.

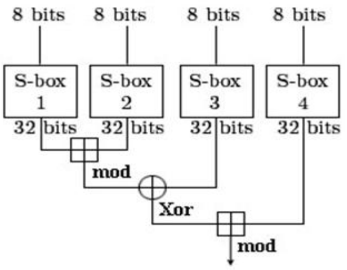


Fig. 11: Round function of Blowfish Algorithm

1. Architecture

Table 2: Architecture [9] [22] [26]

Factors	DES	3DES	IDEA	AES	Blowfish
Developed in year	1974	1998	1991	2000	1993
Developed By	IBM	IBM	Xuejia Lai, James Massey	Rijmen, Daemen	Bruce Schneier
Key length (bits)	56	112, 168	128	128, 192 or 256	32 to 448
Cipher Type	Symmetric Key Algorithm				
Block Size (bits)	64	64	64	128	64
Rounds	16	48	8	10,12,14	16
Used Functions	IP, IP-1, E, P, S-Box(8), PC-1, PC-2	IP, IP-1, E, P, S-Box(8), PC-1, PC-2	N.A	S-Box (1), GF(2^8)	S-Box (4)
Used Operations	XOR, <<<, >>>	XOR, <<<, >>>	XOR, Addition modulo 2^{16}, Multiplication modulo 2^{16+1}	XOR, <<<, >>>	+, XOR, <<<, >>>
Structure of Algorithms	Feistel Network	Feistel Network	Lai-Massey Scheme	Substitution-Permutation Network	Feistel Network
Real World Examples	Digital right management	Blackberry Enterprise server	PGP (Pretty Good Privacy)	Smart Card	SSL

SCALABILITY/ PE

2. Scalability/ Performance

A system is scalable if its performance remains stable even under the critical scenario. The performance of any system can be evaluated on specific criteria. Such criteria's, factors or parameters are known as performance metrics. [6] [11] [15] [27] [31]

Any algorithm can be evaluated on the basis of time and space complexity. To evaluate the performance of encryption and decryption algorithm following criteria's should be considered:-

1. Encryption Time- It can be defined as the time taken to convert a plaintext to a ciphertext
2. Decryption Time- The time Taken for Decryption process. Means time taken for converting cipher text to plain text.
3. Throughput Time- The throughput of the encryption process can be calculated as the total plaintext in bytes encrypted divided by the encryption time.

Total plain text bytes/Total encryption time

4. CPU Utilization & Memory Usage/ Consumption- Total memory used during encryption/ decryption process.
5. Key Length- Total Key length or key size of an algorithm
6. Tunability- It is very popular to define encrypted parts and the encryption parameters used for different applications and requirements.
7. Number of Rounds- Number of iterations in an algorithm.
8. Block Size- The size of each chunk of Plain Text.
9. Number of Functions used- The total number of Functions used for substitution transformation. For example number of S-boxes, Permutation metrics, etc.
10. Number of Operations used – Number of operators like +,*, XOR, <<<, >> etc.
11. Avalanche Effect- It is a required property of cryptographic hashing algorithm. It means that a small change in the input (say one bit) can bring a big change in the output [24] [22].

Avalanche Effect= Number of flipped bits in ciphered text/ Number of bits in ciphered text

- First Order Sac- Change in 1-bit Plain text effect can flip many bits of output
- Higher Order Sac- Change many bits in Plain Text and more than half bits should flip in the output.

1. Robustness against statistical attacks - Statistical analysis is the practice of detecting hidden information by applying statistical tests. Most of the algorithms leave a mark that can be easily detected through statistical analysis. To be able to pass by an eavesdropper without being detected, an algorithm must not leave such a mark in the encrypted file as be statistically significant [9].
2. Robustness against manipulation - In the secure communication, the file may undergo changes by an attacker to remove hidden and encrypted information. It is preferable for the algorithms to be robust against either malicious or unintentional changes [9].
3. Independent of file type - File type and File size also play an important role. There are so many different file types used for communication. The most powerful algorithms thus possess the ability to encrypt any type of information in any type of file [9].

3. Security

Cryptographic security defines whether encryption scheme is secure against Brute Force & different plain - cipher text attacks. A cipher is used to provide protection from undesirable disclosure of plain text. Any cryptanalyst or hacker breaks the cipher with an aim to recover the plaintext. If a secret key is recovered by the hacker, then he can read all messages after that as quickly as the legitimate user. Security is always said relative to threats. Assuming, the attackers have access to everything across the insecure channel is Internet, the security of a cipher can be assessed after considering the computational capability of hacker [10] [13] [14] [16] [20].

Kerckhoff's assumption: The cryptanalyst knows the complete process of encryption and decryption except for the value of the secret key. It implies that the security of a secret-key cipher system rests entirely on the secret key. Following are the few attacks:

- Cipher text-only - Here, the attacker is assumed to have a part of cipher text, and he tries to obtain the associated key to further decrypt the plain text [26].
- Known-plaintext attack – In this we assume the hacker to have few sample of a pair of plain text and cipher text. Using this hacker tries to obtain relative secret key [26].

Table 3: Scalability Criteria's [6] [11] [15] [27] [31]

	DES	3DES	IDEA	AES	Blowfish
Encryption Time	High	High	Low	High	High
Decryption Time	Medium	Medium	High	High	Low

Resource Consumption	Requires more CPU cycles and Memory	Requires effective resource consumption	Lower Power Consumption	Consumes resources when data and block size is big	Requires pre-Processing
Computational Speed	Fast	Moderate	Fast	Fast	Very Fast
Throughput	Medium	Low	High	High	Very High
Avalanche effect	Resists	Resists	Resists	Resists	Resists
Tunability	No	No	No	No	No
Independent of file Type	No	No	No	No	No

Table 4: Security Factors [10][13][14][16][20]

	DES	3DES	IDEA	AES	Blowfish
Plaintext/Ciphertext pairs (Differential Cryptanalysis 2^Block Size)	2^64	2^64	2^128	2^64	2^64
Processing complexity(2^KeySize)	7.2 x 10^16 (2^56)	5.1 x 10^33 (2^112) to 3.74 x 10^50 (2^168)	3.4 x 10^38 (2^ 128) to 1.15 x 10^77(2^256)	3.4 x 10^38 (2^128)	4.29 x 10^9 (2^32) to 7.26 x 10^134 (2^448)
Cryptanalysis Resistance	Vulnerable to Linear and Differential Cryptanalysis, Brute Force	Vulnerable to differential brute force. Attackers can analyze Plain Text, Vulnerable to Chosen plain text, Known plain text	Vulnerable to weak keys(key-schedule attacks and related-key differential timing attacks)	Strong against truncated differential, linear interpolation and square attacks, Vulnerable to Chosen plain text, Known plain text	Vulnerable to differential brute force Attack, Dictionary attack
Security	Inadequate	Vulnerable	High	High	High
Robustness against statistical attacks	Medium	High	High	High	High

- Chosen-plaintext attack – It is an associate attack model. The arbitrary plain text is chosen to be encrypted for the corresponding cipher.4. Chosen-cipher text attack –The cryptologist gathers data, a minimum of partially, by selecting a cipher-text and getting its decipherment beneath an unknown key.
- Chosen-text attack - A chosen text attack is a combination of wanting plain-text and chosen cipher-text attack [9].
- Brute-force attack - This type of attack is a passive attack. This attack performs an exhaustive search for cracking the key. Exhaustive search means calculating every possible combination that could make up a password and test it to see if the password is correct.
- Dictionary attack – In a nutshell, a dictionary attack is a kind of brute force attack where the attacker can rate keys in order of most probable, least likely, compile a list of the most likely (the dictionary), and test them in that order [9].
- Timing attack Timing attacks is a side channel attack which enables an attacker to extract secrets maintained in a security system by observing the time it takes the system to respond to various queries.

- Man-in-the-middle attack - This is the type of active attack. It is a form of eavesdropping in which the attacker controls the entire conversation. Here the attacker secretly relays and possibly alters the communication between two parties who believe they are directly communicating with each other.

4. Flexibility

Table 5: Flexibility [9][26]

	DES	**3DES**	**IDEA**	**AES**	**Blowfish**
Flexibility	No	Yes	No	Yes	Yes
Remarks	No modification is allowed	3 times iteration	No modification is allowed	Structure was flexible to the multiples of 64.	key length must be multiples of 32 bits

5. Limitations (Known Attacks)

Table 6: Limitations [9][22][26]

DES	3DES	IDEA	AES	Blowfish
Highly vulnerable to Linear Cryptanalysis, Weak keys and Brute force. Only 2^56 combinations are required to break the key. It was designed for Hardware so works slow for software.	Exposed to differential and related key attacks. Also susceptible to certain variation of meet-in the middle attack.	Susceptible to different classes of weak keys and are highly exposed to key attacks such as Key Schedule & Related key differential timing attacks. First three rounds of IDEA algorithm is observed for related-key differential timing attacks and Key- schedule attacks.	No serious weakness; Some initial rounds of AES are observed unprotected i.e. initial round can break by square method.	Vulnerable to weak keys, 4 rounds are exposed to 2nd order differential attacks.

5. CONCLUSION

Comparative analysis of few significant block ciphers on different parameters like Security, flexibility, etc. has been done. The architecture and working of all the algorithms have been discussed in the paper. The overall generalized observation being made is that Blowfish algorithm is remarkably better on throughput, power consumption, and processing time as compared to other algorithms. No attack yet has been able to break Blowfish, and it has better performance and efficiency. Another algorithm AES with its variable key length and a varying number of rounds make it an excellent choice to rely on security as well as the flexibility. 3DES is still considered secure besides the only limitation of being extremely slow.

6. REFERENCES

[1] Daemen, J., Govaerts, R. and Vandewalle, J. (1998).*Weak Keys for IDEA*, Springer-Verlag.

[2] Daemen, J., Rijmen, V.(AES Proposal: Rijndael. Retrieved From: http://csrc.nist.gov/encryption/aes/rijndael/Rijndael.pdf

[3] Deep, T.V. & Reddy, V. S. R. (2015). Comparative Analysis of AES finalist algorithms and low power methodology for RC6 block cipher - A Review. International Journal For Technological Research In Engineering, Vol. 2(6)

[4] Denning, D. E. R. (1982). Cryptography and data Security. Addison-Wesley Publishing Company. Menlo Park, California

[5] Ebrahim, M, et al. (2013). *Symmetric Algorithm Survey: A Comparative Analysis*. International Journal of Computer Applications, Vol. 61(20)

[6] Elminaam, A., Abdual, D. S., Kader, H.M., and Hadhoud, M.M. (2010).*Evaluating The Performance of Symmetric Encryption Algorithms*. International Journal of Network Security, Vol.10(*3), PP.213 -219*

[7] Engelfriet, A.(2012). The DES encryption algorithm,‖ Available at www.iusmentis.com/technology/encryption/des/,.

[8] FIPS 197 (2001). Announcing the ADVANCED ENCRYPTION STANDARD (AES). Retrieved From: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[9] Forouzan, B. A., “Data Communications & Networking”, Fourth Edition, 2008, New York: Tata McGraw- Hill.

[10] Jeeva, A.L., Palanisamy, V. & Kanagaram, K.(2012). *Comparative Analysis of Performance Efficiency and Security Measures of Some Encryption Algorithms.* International Journal of Engineering Research and Applications, Vol. 2(3)

[11] Johnson, N. F., Jajodia, S. (1998). Steganalysis of imges created using current steganography software. In IHW’98 -Proceedings of the International Information hiding workshop.

[12] Kak, A. (2015). Computer and Network Security- AES: The Advanced Encryption Standard [pdf file].Retrieved from https://engineering.purdue.edu/kak/compsec/NewLectures/Lecture8.pdf

[13] Kaur, M. Kaur(2016), K. *A Comparative Review on Data Security Challenges in Cloud Computing*. International Research Journal of Engineering and Technology, Vol. 03(1)

[14] LAI, X. (1992). *On the Design and Security of Block Ciphers*, ETH-9752, Zurich

[15] Mehdi, H. (2013). *Checking EABC Performance in Comparison Others Cryptography Algorithms*. International journal of Computer Science & Network Solutions, Vol. 1(1).

[16] Mushtaque, A, M. (2014). Comparative Analysis on Different parameters of Encryption Algorithms for Information Security. International Journal of Computer Sciences and Engineering Vol.-2(4), pp (76-82)

[17] National Bureau of Standards, “Data Encryption Standard,” FIPS Publication 46, 1977.

[18] Nema, P., &.Rizvi, M.A. (2015).Critical Analysis of Various Symmetric Key Cryptographic Algorithms. International Journal on Recent and Innovation Trends in Computing and Communication, Vol.3(6)

[19] Pandav, R.M., & Verma, V.K (2015). *Data Security using Various Cryptography Techniques: A recent Survey.* International Journal for Research in Engineering Application & Management, Vol. 01**(09).**

[20] Rayan, A.M., Abdel-Hafez, A.A., Hafez, I.M. (2016). *Provably Secure Encryption Algorithm based on Feistel Structure*. International Journal of Computer Applications, Vol. 139(1)

[21] Rouse, M. (2006). Data Enryption Standards (DES). Available at http://searchsecurity.techtarget.com/definition/Data-Encryption-Standard.

[22] Schneier B., “Applied Cryptography”, John Wiley& Sons Publication, New York, 1994.

[23] Schneier, B. (1994).Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish),‖ *Fast Software Encryption, Cambridge Security Workshop Proceedings, Springer-Verlag,* 1994, Available at http://www.schneier.com/paper-blowfish-fse.html

[24] Shannon, Claude, Communications theory of Secrecy Systems, Bell systems Technical Journal. 28 (4): 656 - 715.

[25] Sharma, S. & Bhatt, S.(2015). Comparative Analysis - Performance, Efficiency and Security Measures of Block Cipher Algorithms. IJEDR, Vol. 3(3).

[26] Stallings William, “Cryptography and Network Security Principles and Practice”, Fifth Edition, Pearson Education, Prentice Hall, 2011.

[27] Tamimi, A.A. "Performance Analysis of Data Encryption Algorithms. Retrieved October 1, 2008 from http://www.cs.wustl.edu/~jain/cse567-06/ftp/encryption_perf/ index.html

[28] Thaduri, M., Yoo, S. and Gaede, R. “ An Efficient Implementation of IDEA encryption algorithm using VHDL”, Elsevier, **(2004)**.

[29] Tropical Software, Triple DES Encryption, Available at http://www.tropsoft.com/strongenc/des3.htm,

[30] Gatliff, B. (2003). Encrypting data with the Blowfish algorithm. Available at http://www.design-reuse.com/articles/5922/ encrypting-data-with-the-blowfish-algorithm.

[31] Verma, O.P, Agarwal, R., Dafouti, D. & Tyagi, S.(2011). *Performance Analysis of Data Encryption Algorithms.* IEEE- Delhi Technological University India, 2011.

[32] Wagner, R. N. The Laws of Cryptography Retrieved From http://www.cs.utsa.edu/~wagner/laws/