International Journal of Linguistics and Computational Applications (IJLCA) ISSN 2394-6385 (Print)
Volume 3, Issue 2, April– June 2016 ISSN 2394-6393 (Online)

Desperate Attacks and Performance Parameters of Encryption Algorithms

Selin Chandra C S[#1], Karthik M[*2], Saranya A[#3]

[1,3]*Research scholar, PG & Research Dept of Computer science, D.B.Jain College (Autuonomous),Tamil Nadu, India.*
[1]*selin.chandra2@gmail.com,* [3]*saran91aji@gmail.com*
[2]*Assistant Professor, PG & Research Dept of Computer science, D.B.Jain College (Autuonomous),Tamil Nadu, India.*
[2]*karthik2305m@yahoo.co.in*

Abstract— All the aspects of human life are driven by information in the present era. Hence, need to protect useful information from malicious activities such as attacks. Security is playing a vital role in the field of communication system and Internet. Providing security services to the important data under timely manner against attacks are the challenges in the security related communities. Cryptographic ciphers have an important role for providing security to these confidential data against unauthorized attacks. During the practical implementation of these cryptographic ciphers for various applications, there are various factors that can affect the performance and selection of cryptographic algorithms including security which is an important factor. This paper discuss about the desperate attacks and procedures for selecting cryptographic algorithms with respect to performance.

Keywords— Cryptography; Security; Authentication; Attacks; Performance.

1. Introduction

Cryptography is a powerful tool used for the network security. Cryptography algorithms play an important role in information security. Cryptology has been incorporated into smart cards for financial dealings, operating systems, web browsing, mobile phones and electronic identity cards. Cryptography can be divided into Symmetric and Asymmetric key cryptography. There is only one key is used to encrypt and decrypt data in Symmetric key encryption. Symmetric algorithms are also called as secret key algorithms, conventional algorithms, shared algorithms, private key algorithms or one key algorithm.

Two keys are using in Asymmetric key encryption which are known as private keys and public keys. These two keys are related to each other. One key is used for encryption and the other one is used for decryption. Public key is known to the public and private key is known only to the user. Block Ciphers and Stream Ciphers are two types of Symmetric algorithms. The block ciphers are operating on data in groups or blocks. Here the size of the block is of fixed size for encryption. Stream ciphers are operating on a single bit at a time. Here continuous stream is passed for encryption and decryption. Since symmetric encryption requires less computational processing power, they are near to 1000 times faster than asymmetric techniques.

2. Various Cryptographic Algorithms

2.1 Symmetric cryptographic algorithms (Private Key systems)

The private key systems in common use today are,

(a) ROT13

ROT13 is a simple cryptography algorithm which has no key, and it is not secure. [1]

(b) Crypt

The original UNIX encryption program which is modeled on the German Enigma encryption machines. Crypt uses a variable-length key. [1]

(c) DES

The Data Encryption Standard (DES) is an encryption algorithm which is developed by the National Bureau of Standards and Technology and IBM in the 1970s. [1] It was the first encryption standard published by NIST. [2]

DES consists of 16 steps, each of which called as a Round. [5] Many attacks and methods recorded the weaknesses of DES, which made it an insecure block cipher key. [2]

A. Different forms of DES algorithm

(a) DESX

A stronger variation of the DES encryption algorithm is DESX. In DESX, the input plaintext and output is bitwise XORed with 64 bits of additional key material. [4]

(b) DOUBLE DES

It is also called 2DES. Its process is the same as DES but repeated the same process 2 times using two keys K1 and

K2. In the first process, it is produced the cipher text from plaintext using K1 and then take up the cipher text as input, produced another cipher text using K2. [5]

(c) 3DES

As an enhancement of DES, the3DES (Triple DES) encryption standard was proposed. It uses 64 bit block size with 192 bits of key size. The encryption method of 3DES is similar to the one in original DES but applied 3 times for increasing the encryption level and the average safe time. [6]

(d) RC2

RC2 is sold with an implementation that allows keys between 1 and 2048 bits. The RC2mail key length is often limited to 40 bits in software that is sold for export. Unfortunately, a 40-bit key is vulnerable to a brute force attack. [1]

(e) RC4

RC4 is sold with an implementation that allows keys between 1 and 2048 bits. The RC4 key length is often limited to 40 bits in software that is sold for export. Unfortunately, a 40-bit key is vulnerable to a brute force attack. [1]

(f) RC5

RC5 allows a user-defined key length, data block size, and number of encryption rounds. [1]

(g) IDEA

IDEA is used by the popular program PGP to encrypt files and electronic mail. Unfortunately, wider use of IDEA may be hampered by a series of software patents on the algorithm which is currently held by Ascom-Tech AG, in Solothurn, Switzerland.

(h) Skipjack

Skipjack is the algorithm used by the Clipper encryption chip. It uses an 80-bit key. [1]

(i) AES

Advanced Encryption Standard is the new encryption standard recommended by NIST to replace DES. [2][4] The only effective attack known against AES is Brute force attack, in which the attacker tries to test all the characters combinations to unlock the encryption. [11] It is well suited for implementation in hardware and software. [4]

(j) Blowfish

Blowfish is a symmetric key block cipher. It uses a 64 bit block size and variable key length from 32 bits to 448 bits. Blowfish has variants of 14 rounds or less. No attack is known to be successful against it. [2]

(k) Two fish

The Two fish encryption algorithm was designed in order to make the Advanced Encryption Standard (AES). Two fish is a symmetric block code which employs an identical key for encryption and decryption of data. [7]

(l) UMARAM

The UMARAM was designed by Ramesh G and R.Umarani in the year 2010. [6] This algorithm uses a key size of 512-bits to encrypt a plaintext of 512-bits during the 16-rounds. Depending on S-BOX, different shift processes, XOR-Gate, and AND-Gate, a series of transformations have been used in this Algorithm. [6]

(m) UR5

A block encryption algorithm is proposed in this approach. Depending on S-BOX, XOR Gate, and AND Gate, a series of transformations have been used in this Algorithm. The UR5 algorithm encrypts a plaintext of size 64-bits by a key size of 64-bits. It uses eight rounds for encryption or decryption process. It overcomes some drawbacks of the other algorithms. [8]

(n) Camellia

This algorithm specifies the 128-bit block size and 128, 192 and 256 bit key sizes. It was based on feistel network cipher with 18 or 24 rounds. [8]

2.2 Asymmetric cryptographic algorithms (Public key systems)

The public key systems in common use today are,

(a) Diffie-Hellman

. It is not actually a method of encryption and decryption, but a system for developing and exchanging a shared private key over a public communications channel. After the two parties agree to some common numerical values, each party creates a key. They are exchanging the Mathematical transformations of the keys. There are existing several versions of this protocol. [1]

(b) RSA

The well-known public key cryptography system developed by MIT professors Ronald Rivest and Adi

Shamir, and by USC professor Leonard Adleman in 1977. [1] RSA algorithm is the most popular and proven asymmetric key cryptographic algorithm. [5] RSA can be used both for encrypting information and as the basis of a digital signature system. To prove the authorship and authenticity of digital information, Digital signatures can be used. Depending on the particular implementation used, key length is varying. If the keys are longer, then it is more secure. [1]

The RSA algorithm is based on the mathematical fact that it is easy to find and multiply large prime numbers together. [5]

(c) ElGamal

This algorithm is based on exponentiation and modular arithmetic. It may be used for encryption and digital signatures in a manner similar to the RSA algorithm. If the keys are longer, then it is considered to be more secure. [1]

(d) DSA

The Digital Signature Algorithm developed by NSA. It is adopted as a Federal Information Processing Standard (FIPS) by NIST. The DSA key may be any length. However, only keys between 512 and 1024 bits are permitted under the FIPS. [1]

3. Desperate Attacks

3.1 Classification of Attacks

Two classes of attack: Cryptanalytic attacks and Implementation attacks. The former tries to attack mathematical weaknesses in the algorithms whereas the latter tries to attack the specific implementation of the cipher (such as a smartcard system). An attack can be passive or active based on the action performed by the attacker. Thus, the classifications of attacks become passive or active.

A. Passive Attacks

If there is an unauthorized access to the information, then that attack is known as passive attack. For example, actions such as blocking and listen secretly to a private conversation on the communication channel can be regarded as passive attack. Since they neither affect information nor disrupt the communication channel, these actions are passive in nature. A passive attack is often seen as stealing information. Theft of data still leaves the owner in possession of that data. This is the only difference in stealing physical goods and stealing information. Passive information attack is thus more dangerous than stealing of goods, as information theft may go unnoticed by the owner. [12]

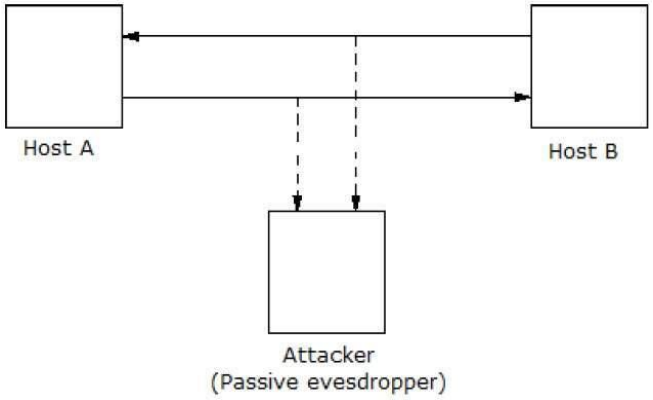

Fig.1: Passive Attacks

B. Active Attacks

Changing the information in some way by conducting some process on the information is known as an active attack. For example,

- Information is modified in an unauthorized manner.
- Initiating unintended or unauthorized transmission of information.
- Alteration of authentication data such as originator name or timestamp associated with information
- Deletion of data in an unauthorized way.
- Denial of access to information for legitimate users (denial of service).

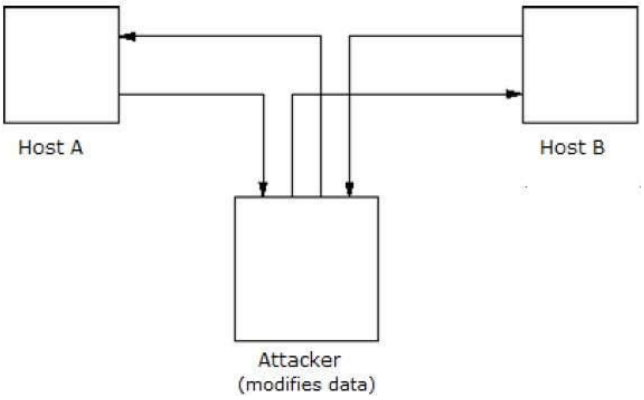

Fig. 2: Active Attacks

Cryptography provides many tools and techniques to prevent the data against most of the attacks [12].

3.2 Types of cryptanalytic attack

Based on the methodology used, attacks on cryptosystems are categorized as,

Ciphertext-only attack: In this attack, the cryptanalyst obtains a sample of cipher text, but the plaintext did not associate with it. The attacker will try to find the key or decrypt one or more pieces of cipher text. One will get this data easily in many scenario. However, without a very large cipher text sample a successful ciphertext-only attack is difficult.

Known-plaintext attack: In this type of attack the cryptanalyst gets a sample of ciphertext and the corresponding plaintext as well.

chosen-plaintext attack: In chosen-plaintext attack the cryptanalyst can choose a quantity of plaintext and then obtain the corresponding encrypted ciphertext.

Adaptive-chosen-plaintext attack: It is a special case of chosen-plaintext attack. In this attack, the cryptanalyst choose plaintext samples dynamically, and alter his or her choices based on the results of previous encryptions.

Chosen-ciphertext attack: Here, the cryptanalyst may choose a piece of ciphertext and tries to obtain the corresponding decrypted plaintext. This type of attack is mostly in public-key cryptosystems.

Aadaptive-chosen-ciphertext: It is the adaptive version of the above attack. In this attack, a cryptanalyst has free use of a piece of decryption hardware, but is unable to extract the decryption key from it.

- *Dictionary Attack* − This attack involves compiling a 'dictionary'. An attacker is learning cipher text and plain text over a period of time. Then he builds a dictionary which contains cipher text and corresponding plaintexts that he has learnt. In future, when an attacker gets the ciphertext, he refers the dictionary to find the corresponding plaintext. [12]
- *Brute Force Attack (BFA)* − in this method, the attacker tries to determine the key by attempting all possible keys. If the key is 8 bits long, then the number of possible keys is $2^8 = 256$. If the attacker knows the ciphertext and the algorithm, he tries all the 256 keys one by one to decrypt cipher text in to corresponding plain text. If the key is long, the time to complete the attack would be very high. [12]
- *Birthday Attack* − This attack is a variant of brute-force technique. It is used against the cryptographic hash function. If the attacker is able to find two different inputs that give the same hash value, it is a **collision** and that hash function is said to be broken. [12]
- *Man in Middle Attack (MIM)* − The targets of this attack are mostly public key cryptosystems where key exchange is involved before communication takes place.
  - Host *A* wants to communicate to host *B*, hence requests public key of *B*.
  - An attacker intercepts this request and sends his public key instead.
  - Thus, whatever host *A* sends to host *B*, the attacker is able to read.
  - The attacker re-encrypts the data after reading with his public key and sends to *B*, in order to maintain communication
  - The attacker sends his public key as *A*'s public key. So *B* takes it as if it is taking it from *A*. [12]
- *Side Channel Attack (SCA)* − *Side Channel Attack* is not against any particular type of cryptosystem or algorithm. Instead, it is launched to exploit the weakness in physical implementation of the cryptosystem. [12] A side channel attack influences additional information, such as CPU cycles used(or time taken), to perform a calculation, voltage used, and so on.

TEMPEST: TEMPEST attacks involve the remote or external detection and collection of the electromagnetic signals emitted from a cryptographic module during processing. TEMPEST attack can be used for getting keystroke information, messages displayed on a video screen, and other forms of critical security information (e.g., cryptographic keys).

- *Timing Attacks* –Depends on the timing difference of various computations, (the fact that different computations take different times to compute on processor) the attacker tries to know about a particular computation the processor is carrying out. For example, if the encryption takes a longer time, it indicates that the secret key is long. [12]
- *Power Analysis Attacks* − Power Analysis Attacks are similar to timing attacks except that the amount of power consumption is used to obtain information about the nature of the underlying computations. [12]
- *Fault analysis Attacks* − In these attacks, errors are induced in the cryptosystem and the attacker studies the resulting output for useful information. [12]

SSL MITM attack - In this type, the attacker becomes a successful man-in-the-middle connection by intruding into the network. Attacker silently observes the HTTPS traffic on the wire, and waits for the targeted website to respond to some browser's HTTPS request. The server is supposed to send its digital certificate to browser as a part of SSL handshake process. Attacker grabs this certificate, and notes down various details such as domain name, expiration date, cipher strength etc. A self-signed certificate is created by an Attacker as his own certificate. It contains the same information as that of the captured certificate. From here itself, attacker becomes a real man-in-the-middle. He intercepted browser request and responded with the fake certificate. [11]

SSL MITB attack – In this type of attack, to create a man-in-the-browser situation, attacker injects a JavaScript code snippet into the browser. This snippet monitors all SSL activities and records the session. While this is happening, the attacker also records encrypted version of the same session and programmatically tries to find out cipher strength and the key, besides stealing data. This attack is becoming more popular lately, due to multiple open source

browsers and various security vulnerability problems with each of those. [11]

Key Hijacking - This is an intrusive type of attack. Here the attacker gains access into the web server. Web server is one which hosts the website. When the server is compromised, the private key can be obtained by the attacker uses elevated privilege attack to gain access to the certificate store. To download an entire HTTPS session, an attacker is using packet sniffing. Then, it stores for offline decryption. Using the private key which is already stolen, performing decryption process. The public key is available in the browsers trusted authority key store or public files. The data set deciphered so, might reveal vital personal information such as userid, address, credit card number etc, assuming that the targeted website sells goods online using e-commerce technology. [11]

Birthday SSL Attack - This attack depends on a mathematical theory called as birthday problem paradox. Birthday problem paradox says that some pairs of people will have same birthday. These pairs of people are from a set of randomly selected people. If the number of people chosen is large, this theory becomes more accurate. Multiple attackers coming together and targets on the hash in the Birthday attack. They share the chunks of data which are captured individually. Each chunk is then analyzed programmatically to create additional set of data, in such a way that the hash of it matches that of the data chunk. Further process of the original data chunk and the resultant data set, helps derive the encryption key. Birthday attack is a very time consuming. It is technically complex type .However, by using multiple powerful computing machines and software programs, Birthday attack can be possible. [11]

Chosen Dataset attacks –In Chosen Dataset attacks**,** attackers always aim for data as well as the key in order to completely compromise a cryptographic system. There are two types of methods in a chosen dataset.(1) Chosen plain text (2) Chosen cipher text In Chosen plain text method, attacker is assumed to have access to the original data and the encrypted version of it(i.e., plain text and cipher text). Attacker then applies multiple encryption keys to the original datat to produce cipher text, each time the output is compared with the already encrypted version. If the result is positive, it means the key is derived. In the Chosen cipher text, attacker has the cipher text and also the decrypted version of it. Again, attacker tries multiple keys until the output matches that of the decrypted version obtained already. These attacks are bit less time consuming. To seek the desired results, attacker need enormous amount of data and computational power. [11]

SSL Brute force attack - This is a different type of attack. In *SSL Brute force attack*, attacker sends very small data sets to be encrypted by SSL protocol. Attacker captures the resultant outcome and stores it against the transmitted dataset. A key can be eventually derived by performing such operation on lots of data chunks. This process is very slow. It can take days to decipher the key. The origin of this attack is found to be from within the firm's network. To speed up the process this method is usually combined with the group key decipher attack. [11]

Group Key Deciphering - key based encryption is dependent on the length of key, where a bigger key result into lot of time required deciphering it. In group key deciphering attack, multiple attackers come together, each one with their powerful machine. Only a given set of data is captured and used in group method. This data is subjected to all the possible permutations of keys, to try decrypting the data. Normally a 256bit encryption can take multiple years to decipher. But the usage of multiple powerful computing machines can bring the time down. To bring that time down further, Attackers also use statistical grouping of keys to be tried from different machines. In past, few such experiments showed that cracking a 128bit key required only few days. It is feared that cracking a 1024bit key can unfortunately be a reality soon with improving CPU speeds and throughputs. [11]

Compromised key attack - A trusted certificate provider authority signs a certificate, so Cryptography is all about trust. The provider itself is supposed to be extremely secure. But in the past, it has unfortunately happened that trusted certificate provider's private key is either exposed or stolen by attacker. By using this private key, attacker signs certificate created for a domain name, which is their own site. Any browser being attracted to this website will not suspect such a website; this is because the certificate will pass the authenticity test. This happens because the public key of such certificates will already be present in the browser certificate store. This can, and in the past has, resulted into loss of personal information. [11]

SSL DoS - Attackers main aim is usually to steal the data. Since it is a troublesome and highly technical process in cryptography, few attackers tend to use legacy methods such as a denial of service attack. To slow down the communication for achieving security, SSL negotiation adds it payload on the TCP protocol. To achieve SSL denial of service attack, the attacker establishes SSL communication through a browser and then sends multiple false packets with varying length on that channel. Each packet is decrypted and processed on the server side, thus eventually exhausting CPU power, resulting into service outage. In another form which takes place at layer-3, the TCP port 443 is attacked with false fragmented packets, creating similar effect. [11]

4. Measuring the Parameters and Analysing the Performance of Algorithms

One of the important components of any encryption algorithm is Performance. This part gives a description about simulation environment, system components, and various metrics for the performance and the procedure for

analyzing the performance of algorithms.

(a) Different System parameters

Performance of algorithm can be analyzed by,

- Using 2 architectures such as wired architecture and wireless architecture.
- Using different system configuration such as laptop, standalone pc, Networked pc to get better comparison results.
- Using Different operating systems.

(b) Various Metrics For The Performance

Evaluation of the performance of proposed algorithm is based on the several various metrics which are best suited for the cryptographic algorithms. Some selected metrics for the evaluation are Encryption time, decryption time, Throughput of encryption, Throughput of decryption, CPU process time, CPU clock cycles Power consumption, Memory Utilization.

Encryption time is the total time taken by an algorithm to produce a cipher text from plain text. It is used to calculate the throughput of the encrypted algorithm. It gives the rate of encryption. [10] Decryption time is the time taken by an algorithm to produce plain text from cipher text. It is used to calculate the throughput of the decrypted algorithm. It gives the rate of decryption. [10] The speed of encryption is calculated from throughput of the encryption. [10] Throughput of encryption = Plain Text (MB) / Encryption time. [2] Throughput of the encryption algorithm and the power consumption algorithm are inversely proportional to each other. If there is an increase in the throughput of the encryption algorithm, there is a decrease in the power consumption algorithm. [10]

The speed of decryption is calculated from throughput of the decryption. [10] Throughput of decryption = Plain Text (MB) / Encryption time. [2] Throughput of the decryption algorithm and the power consumption algorithm are inversely proportional to each other. If there is an increase in the throughput of the decryption algorithm, there is a decrease in the power consumption algorithm. [10] The CPU process time is the time that a CPU is committed only to the particular process of calculations. It reflects the load of the CPU [9]. More the CPU time used in the encryption process, the higher is the CPU load. [10] The CPU clock cycles are a metric which is reflecting the energy consumption of the CPU while operating on encryption operations. Each cycle of CPU will consume a small amount of energy.

Power consumption is the total power that required by the encryption and decryption algorithm. It was estimated based on the throughput of the encryption and decryption algorithm. When increase in the throughput of the encryption/decryption algorithm, there is a decrease in the power consumption algorithm. [10] Memory Utilization is the analysis of memory requirement for the encryption and decryption. [10]

The formula to calculate the average encryption time

$$AvgTime = \frac{1}{Nb}\sum_{i=1}^{Nb}\frac{M_i}{t_i}\ (Kb/s)$$

AvgTime=Average Data Rate(Kb/s)
Nb=Number of Messages
Mi=Message size(Kb)
ti=Time taken to Encrypt Message Mi

Energy consumption for encryption and decryption can be measured in many ways. [9] [10] These methods as follows: The First method used to measure energy consumption is to assume that an average amount of energy is consumed by normal operations and to test the extra energy consumed by an encryption algorithms. This method simply monitors the level of the percentage of remaining battery that can computed by equations
The battery life consumed in percentage for one run = Change_in_Batterylife / No_of_Runs

Average battery Consumed per iteration = $\sum_{1}^{N}\frac{\text{Battery\_consumed/Iteration}}{\text{No\_of\_Runs}}$

The second method of security primitives can also be measured by counting the amount of computing cycles which are used in computations related to cryptographic operations. For computation of the energy cost of encryption, we use the same techniques as described in using the following equations.

Bcost_encryption (ampere-cycle)= τ * I

$$Tenergy\_cost\ (ampere - Seconds) = \frac{B\_cost\_encryption}{F(\frac{Cycles}{Sec})}$$

Ecost(Joule) = Tenergy_cost (ampere-seconds) * V
Where Bcost-encryption: a basic cost of Encryption
(ampere-cycle): The total number of clockcycles.
I: the average current drawn by each CPU clock cycle
Tenergy_cost: The total energy cost(ampere-seconds),
F: Clcok frequency(cycles/sec)
Ecost(joule): the energy cost(consumed)

By using the cycles, the operating voltage of the CPU, and the average current drawn for each cycle, we can calculate the energy consumption of cryptographic functions. So, the amount of energy consumed by program P to achieve its goal (encryption or decryption) is given by

E= VCC * I * N * τ

Where N – The number of clock cycles and VCC – The supply voltage of the system
I – the average current in amperes drawn from the power source for T seconds.

(c) Procedure For Analysing Performance

Several experimental procedures are used for analyzing performance.

- Files of same type with different packet size: performance are compared by encrypting input files of varying sizes and their encryption time is calculated, [2] for example : Different file sizes ranging from 40 Kb to 8000Kb. [10]
- Files with different Data types: [3] Different data type files like audio, image, textual and video of same file size are chosen and encryption time of different cipher algorithms is calculated for these data types. For all executions of a specific cipher algorithm, varying parameter is data and constant parameter is key size.
- Different types of files based on varying key size: [4] Different data type files like audio, image, textual and video of same file size and varying key size are chosen and encryption time of different cipher algorithms is calculated for these data types, for example .exe(Executable file), .doc(document file), .wmv(Window Media Video), avi (Audio Video Interface).
- File with different data densities: [3] This study is taken to check whether any effect on the encryption time depends on density of data or not. Encryption rate is evaluated for the two different data density file For a cipher algorithm, key size and block mode are kept at bare minimal parameters, for example a sparse file of 69MB and a dense file of 58.5MB.
- Encryption Algorithms with different key sizes: Different key sizes are used to analyze the performance of the selected algorithms. [10] This study will analyze the effect of changing the size of encryption key on encryption time, for example BMP file of 50.5MB is taken and different cipher algorithms are executed for different size of keys supported. [3]
- Performance of algorithm in different web browsers: In this study, one can use different Web browsers like Internet Explorer, Mozilla Firefox, Opera, Netscape Navigator and Google Chrome in order to determine which type of algorithm is suitable to which type of Web browser in terms of their performance and compatibility. [6]

(d) Implementation

The algorithm's simulation program can be implement using in one of the languages such as JAVA, [2] C#, [7] ASP. [6]

5. Purpose for Analyzing the Performance of Algorithms

Encryption makes the modern world go round. Every time we make a mobile phone call, buy something with a credit card in a shop or on the web, or even get cash from an ATM, encryption bestows upon that transaction the confidentiality and security to make it possible.

If only the encryption algorithm cryptographically strong, it can be used in secure transaction.

Cryptographically strong means that the described method has some kind of maturity and approved for use against various systematic attacks in theory and practice. Indeed, that the method may resist those attacks long enough to protect the information carried for a useful length of time.

Attacks are depends upon the many factors such as speed of the algorithm, key management .In some algorithms, if the size of the key is so huge it is impossible for an attacker to search through the key space with the resources they usually have . In some other algorithms, if the time (encryption/decryption) taken for the algorithm is less, the cipher cannot be broken.

Depends on the study of possible attacks and performance of algorithm with respect to the key size, packet size, data type, data density and web browser , one can choose the algorithm which is Cryptographically strong .

6. Conclusion and Future Scope

This paper provides the small description of various attacks and the methodology for analyzing performance of algorithms. In future, performance evaluation of selected cryptographic symmetric and asymmetric algorithms can be done by using these techniques to strengthen the security procedure and improve the speed.

References

[1] Cryptography in Practical UNIX and Internet Security http://www.diablotin.com/librairie / networking /puis/ch06_04.htm.

[2] Pratap Chandra Mandal "Evaluation of performance of the Symmetric Key Algorithms: DES, 3DES ,AES and Blowfish" in Journal of Global Research in Computer Science(Volume 3, No. 8, August 2012) ISSN-2229-371X.

[3] Ranjeet Masram, Vivek Shahare, Jibi Abraham, Rajni Moona "Analysis And Comparison Of Symmetric Keycryptographic Algorithms Based On Various File Features" in International Journal of Network Security & Its Applications in (IJNSA), Vol.6, No.4, July 2014.

[4] Rishabh Arora, Sandeep Sharma "Performance Analysis of Cryptography Algorithms" in International Journal of Computer Applications (0975 – 8887) Volume 48– No.21, June 2012.

[5] Sombir Singh, Sunil K Maakar and Dr. Sudesh Kumar "A Performance Analysis of DES and RSA Cryptography" International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)Web Site: www.ijettcs.org Email: editor@ijettcs.org, ditorijettcs@gmail.com Volume 2, Issue 3, May – June 2013 (ISSN 2278-6856).

[6] G. Ramesh, R. Umarani Performance Analysis of Most Common Encryption Algorithms on Different Web Browsers in I.J. Information Technology and Computer Science, 2012, 12, 60-66 Published Online November 2012 in MECS (http://www.mecs-press.org/) DOI: 10.5815/ijitcs.2012.12.06.

[7] Lalit Singh, Dr. R.K. Bharti " Comparative Performance Analysis of Cryptographic Algorithms" International Journal of Advanced Research in Computer Science and Software Engineering in Volume 3, Issue 11, November 2013 (ISSN: 2277 128X).

[8] M.Anand kumar, S. Umadevi "Comparative analysis of symmetric encryption algorithms for data communication" in Karpagam JCS in Volume 7, Issue 5,July-Aug 2013(ISSN 0973-2926).

[9] G. Ramesh1 Dr. R. Umarani "Performance Analysis of Most Common Symmetrical Encryption Algorithms" International Journal of Power Control Signal and Computation(IJPCSC), Vol3. No1. Jan-Mar 2012 ISSN: 0976-268X www.ijcns.com.

[10] M.Anand kumar and K.Appathurai "Performance analysis of Blow Fish, IDEA and AES Encryption algorithms in Karpagam JCS in Volume 9, Issue 1,Nov-Dec 2014(ISSN 0973-2926).

[11] http://www.valencynetworks.com/ articles/ cyber- attacks - cryptographic- attacks.html.

[12] http:// www.tutorialspoint.com / cryptography/ attacks_ on _ cryptosystems.htm

[13] Divya.R#1, Sowmiya.R, "Cryptography in Digital Watermarking - A Wavelet based Contrast Sensitivity", Special Issue of Engineering and Scientific International Journal, (TSRW-MCA-SAEC) – May 2015, pp.73-78.