International Journal of Computer Science and Engineering Open Access
Research Paper Volume-2, Issue-3 E-ISSN: 2347-2693

Triple Security of File System for Cloud Computing

Richa Arya

Department of Computer Science and Engineering, NGF college of Engineering & Technology, Palwal
ricsarya@gmail.com

www.ijcseonline.org

Received: 7 March 2014 Revised: 14 March 2014 Accepted: 26 March 2014 Published: 31 March 2014

Abstract- Large scale distributed systems such as cloud computing applications are becoming very common. These applications come with increasing challenges on how to transfer and where to store and compute data. The most prevalent distributed file systems to deal with these challenges is the Hadoop File System (HDFS) which is a variant of the Google File System (GFS). However HDFS has two potential problems. The first one is that it depends on a single name node to manage almost all operations of every data block in the file system. As a result it can be a bottleneck resource and a single point of failure. The second potential problem with HDFS is that it depends on TCP to transfer data. As has been cited in many studies TCP takes many rounds before it can send at the full capacity of the links in the cloud. This results in low link utilization and longer download times. To overcome these problems of HDFS we present a new distributed file system. Our scheme uses a light weight front end server to connect all requests with many name nodes i.e Triple Security.

Keywords : Network, Performance, Security, Signature

I. INTRODUCTION

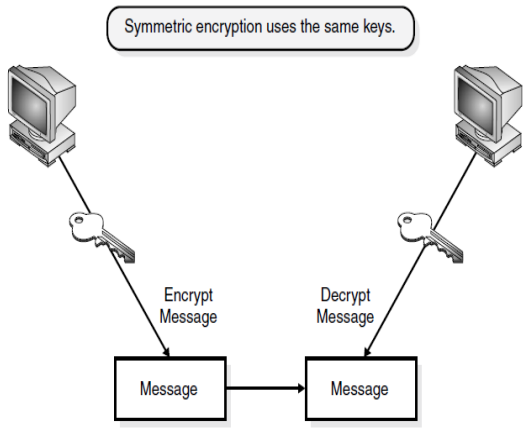
CRYPTOGRAPHY

Cryptography is a method of storing and transmitting data in a form that only those it is intended for can read and process. It is a science of protecting information by encoding it into an unreadable format. Cryptography is an effective way of protecting sensitive information as it is stored on media or transmitted through network communication paths. Although the ultimate goal of cryptography, and the mechanisms that make it up, is to hide information from unauthorized individuals, most algorithms can be broken and the information can be revealed if the attacker has enough time, desire, and resources. So a more realistic goal of cryptography is to make obtaining the information too work-intensive to be worth it to the attacker.

The process of encryption transforms plaintext into ciphertext and the process of decryption transforms ciphertext into plaintext.

Symmetric Systems

There are several types of symmetric algorithms used today. They have different methods of providing encryption and decryption functionality. The one thing they all have in common is that they are symmetric algorithms, meaning two identical keys are used to encrypt and decrypt the data.



II. RELATED WORKS

DSA (Digital Signature Algorithm)

A digital signature algorithm authenticates the integrity of the signed data and the identity of the signatory. A digital signature algorithm may also be used in proving to a third party that data was actually signed by the generator of the signature. Is intended for use in electronic mail, electronic data interchange, software distribution, and other applications that require data integrity assurance and data origin authentication. The wireless protocols, like Hiper LAN/2, and WAP, have specified security layers and the digital signature algorithm have been applied for the authentication purposes. Electronic Signature can prove the Authenticity of Alice as a sender of the message.

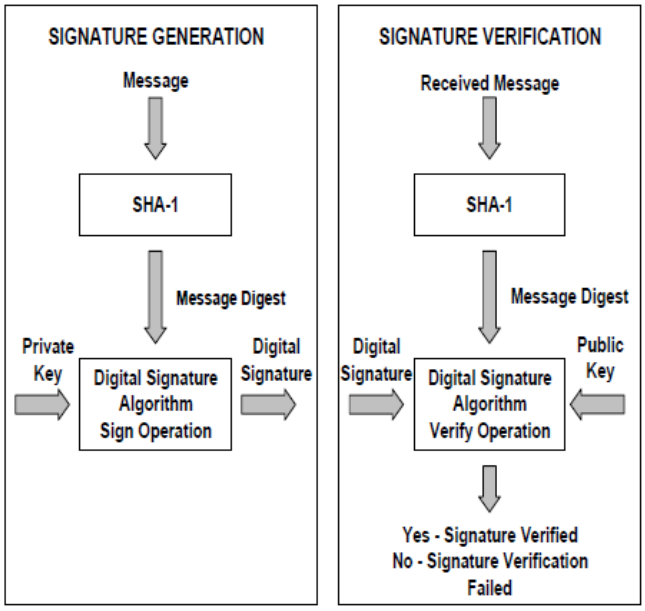
Corresponding Author: *Richa Arya*

The Digital Signature Standard (DSS) uses three algorithms for digital signature generation and verification [1,3]. The Digital Signature Algorithm (DSA), the RSA digital signature algorithm as defined in ANSI X9.31 and Elliptic Curve digital signature algorithm (ECDSA) as defined in ANSI X9.62.

DIGITAL SIGNATURE ALGORITHM

A digital signature is represented in a computer as a string of binary digits. A digital signature is computed using a set of parameters and authenticates the integrity of the signed data and the identity of the signatory. An algorithm provides the capability to generate and verify signature. Signature generation makes use of a private key to generate a digital signature. Signature verification makes use of a public key, which corresponds to, but is not the same as, the private key. Each user possesses a private and public key pair. Public keys are assumed to be known to the public in general. Private keys are never shared. Anyone can verify the signature of a user by employing that user public key. Only the possessor of the user private key can perform signature generation.

A hash function is used in the signature generation process to obtain a condensed version of data, called a message digest. The message digest is then input to the digital signature algorithm to generate the digital signature. The digital signature is sent to the intended verifier along with the message. The verifier of the message and signature verifies the signature by using the sender's public key.



DES (Digital Encryption Standard)

DES was designed by IBM and adopted by the U.S.govt.as the standard encryption method. The Data Encryption Standard (DES) is a block cipher that uses shared secret encryption and Uses only a single key.

S-DES encryption (decryption) algorithm takes 8-bit block of plaintext (cipher text) and a 10-bit key, and produces 8-bit cipher text (plaintext) block. Encryption algorithm involves 5 functions: an initial permutation (IP); a complex function fK, which involves both permutation and substitution and depends on a key input; a simple permutation function that switches (SW) the 2 halves of the data; the function fK again; and finally, a permutation 2 function that is the inverse of the initial permutation (IP-1). Decryption process is similar. The function fK takes 8-bit key which is obtained from the 10-bit initial one two times. The key is first subjected to a permutation P10. Then a shift operation is performed. The output of the shift operation then passes through a permutation function that produces an 8-bit output (P8) for the first sub key (K1). The output of the shift operation also feeds into another shift and another instance of P8 to produce the 2nd sub key K2. We can express encryption algorithm as superposition:

$$IP^{-1} \circ f_{K_2} \circ SW \circ f_{K_1} \circ IP$$

or

$$\text{Ciphertext} = IP^{-1}\left(f_{K_2}(SW(f_{K_1}(IP(plaintext)))))\right)$$

Where

$$K_1 = P8(Shift(P10(key)))$$

$$K_2 = P8(Shift(Shift(P10(key))))$$

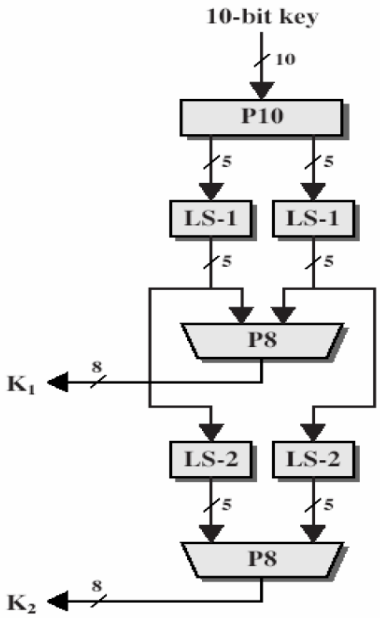
Decryption is the reverse of encryption:

$$\text{Plaintext} = IP^{-1}\left(f_{K_1}(SW(f_{K_2}(IP(ciphertext)))))\right)$$

We now examine S-DES in more details

DES key Generation.

Scheme of key generation:



Simplified DES key Generation.

Advanced Encryption Standard (AES)[7-10]

After DES was used as an encryption standard for over 20 years and it was able to be cracked in a relative short amount

of time, NIST decided a new standard, the Advanced Encryption Standard (AES), needed to be put into place. This decision was announced in January 1997, and a request for AES candidates was made. The AES was to be a symmetric block cipher algorithm supporting keys sizes of 128-, 192-, and 256-bit keys.
The following five algorithms were the finalists:
• MARS Developed by the IBM team that developed Lucifer
• RC6 Developed by the RSA Laboratories
• Serpent Developed by Ross Anderson, Eli Biham, and Lars Knudsen
• Two fish Developed by Counterpane Systems
• Rijndael Developed by Joan Daemon and Vincent Rijmen.

2.1.3)Rivest Shamir Ad lEman (RSA):

The RSA algorithm can be used for both public key encryption and digital signatures. Its security is based on the difficulty of factoring large integers.

ENCRYPTION OF RSA:

1. Obtains the recipient B's public key (n, e).
2. Represents the plaintext message as a positive integer *m*.
3. Computes the cipher text c = me mod n.
4. Sends the cipher text *c* to B.

DECRYPTION OF RSA:

1. Uses his private key (n, d) to compute m = cd mod n.
2. Extracts the plaintext from the message representative m.

2.2 STEGANOGRAPHY

2.2.1Introduction to Steganography

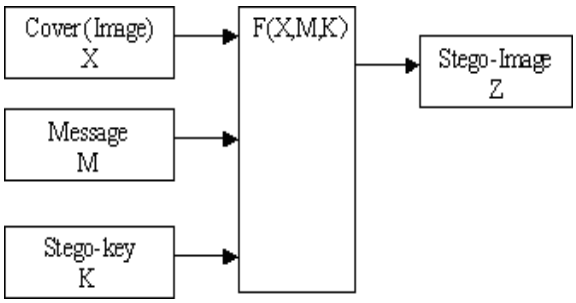
Steganography is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message. Steganography is the process of hiding one medium of communication (text, sound or image) within another. The word Steganography comes from the Greek steganos (covered or secret) and graphy (writing or drawing) and so it literally means, covered writing.

Steganography is the practice of encoding secret information in a manner such that the very existence of the information is concealed under the image or picture where it is hided. Throughout history, many steganography techniques have been documented, including the use of cleverly-chosen words, invisible ink written between lines, modulation of line or word spacing, and microdots. Usually the secret information is concealed by the use of an innocuous cover so as to arouse no suspicion to anyone. As an example, the cover text: “I’m feeling really stuffy. Emily’s medicine wasn’t strong enough without another febrifuge”.

Digital Steganography

Steganography is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message [1]. Digital Steganography deals with developing and transmitting digital data/files under the cover of image/pictures. A typical digital steganography encoder is shown on Figure 1.1. The message is the data that the sender wishes to remain confidential and can be text, images, audio, video, or any other data that can be represented by a stream of bits. The cover or host is the medium in which the message is embedded and serves to hide the presence of the message. This is also referred to as the message wrapper. The message embedding technique is strongly dependent on the structure of the cover, and here in this thesis report covers are restricted to being digital images. It is not required that the cover and the message have homogeneous structure. For example, it is possible to embed a recording of Shakespeare’s lines (an audio stream message) inside a digital portrait of the famous playwright (an image cover).

The image with the secretly embedded message produced by the encoder is the stego-image. The stego image should resemble the cover image under casual inspection and analysis. In addition, the encoder usually employs a stego-key which ensures that only recipients who know the corresponding decoding key will be able to extract the message from a stego-image.



Steganography Encoding

Recovering the message from a stego-image requires the stego-image itself and a corresponding decoding key if a stego-key was used during the encoding process. The original cover image may or may not be required; in most applications it is desirable that the cover image is not needed to extract the message. Steganography is not the same as cryptography. In cryptography, the structure of a message is changed to render it meaningless and unintelligible unless the decryption key is available. Cryptography makes no attempt to disguise or hide the encoded message. Steganography does not alter the structure of the secret

message, but hides it inside a cover. It is possible to combine the techniques by encrypting a message using cryptography and then hiding the encrypted message using steganography. The resulting stego-image can be transmitted without revealing that secret information is being exchanged.

2.2.1 Classification of Steganography Techniques
Steganography means the concealment of the secret data by the use of the innocuous cover so as to arouse no suspicion even if hostile agents discover the cover. So depending upon how the secret data is concealed behind the cover medium, the steganography technique [3] can be categorized into three broad categories:

1. Technical steganography
2. Linguistic steganography and
3. Data steganography.

Data Steganography

Data Steganography is that branch in which we use the digital data as the carrier to embed the secret data. The start of computers has allowed us to begin embedding messages into pictures or sound files. To the human eye, the picture itself remains unchanged, yet within it there could be up to a book's worth of information. We explain how it is achieved. Computers operate in binary and every letter, data and instruction has binary code. For example the binary code of "237" is 11101101.The last digit of all binary messages which is underlined in the example, is neither meaningful nor necessary, is known as the Least Significant Bit (LSB). The last bit or the LSB is least important because it has the least numerical value. Let us explain the concept by citing the following example:

The binary code of "237" is: Binary code of "237":11101101 Now if we change the LSB of "237" binary code i.e. from "1" to "0" then the changed binary code of "237" will become "1110 1100".So the decimal value will become 236. Code of "237" after replacement of bit: 1110 1100. The decimal value will change from "237" to "236" after replacement. As by modifying the LSB there is only change of 1 in the numerical value of the digital data which is not detected by the human eyes under the naked eye so LSB is used for embedding the secret data.

In digital medium such as images, audio files and video files the change in the value of their binary code by "1" has no significant effect. So the LSB of the digital data can be used in data Steganography to hide the secret data. The possible digital carrier or data used in steganography are images, text, audio, and video. We will explain how these digital medium are used.

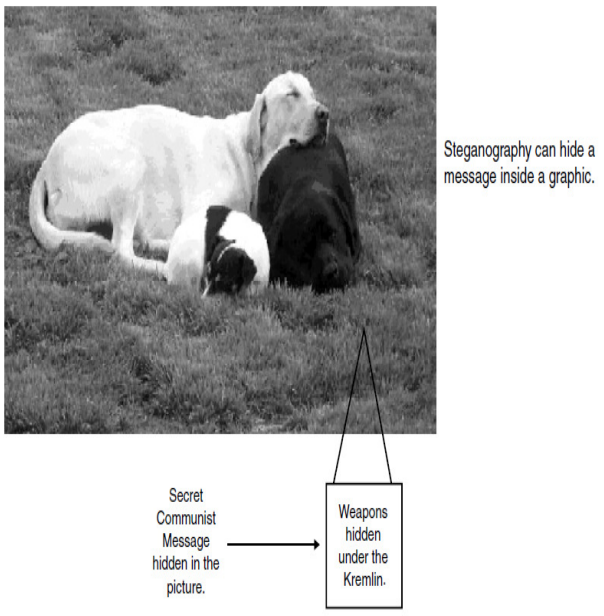
i) Image Steganography(Hiding Data in Images)
Image steganography techniques can be classified on the basis of the domains in which data is embedded. Basically there are two domains, the spatial domain and the transform domain. Steganographic techniques try to embed data in these domains.
In the spatial domain [4] image steganography the simplest technique is to embed data in the least significant bit (LSB) of each pixel in the cover image. The LSB Replacement technique alters the insignificant information in the cover image. It places the embedding data at the least significant bit (LSB) of each pixel in the cover image. There are two types of LSB insertion methods; fixed-sized and variable-sized. The former embeds the same number of message bits in each pixel of the cover-image whereas the latter embeds a random number of bits per pixel.

In the transform domain data can be hidden by modifying the Discrete Cosine Transform (DCT) coefficient values. A 24-bit image has 3 byte, one for each of the three primary color values (red, green and blue) of each pixel. If we consider just the blue there will be 28 different values of blue. The difference between say 11111111 and 11111110 in the value for blue intensity is likely to be undetectable by the human eye. Therefore, the least significant bit can be used for something else other than color information. If we do it with the green and the red as well we can get one latter of ASCII text for every three pixels we take a 8 bit image file by which we mean that 8 bits are required to store the color value of one pixel of a image.

For example, we had the following values of a 8 bit image file: The color values of the pixel of cover image in the decimal form are 132 134 137 141 121 101 74 38 In binary, those values would be represented as:

Binary form of pixels of carrier image: 10000100 10000110 10001001 10001101 0111001 01100101 01001010 00100110 Binary code of secret message: 10010101 (149) Now we wanted to hide the binary file 10010101(149) inside the carrier data. We simply replace the least significant bit of each pixel's binary value (the last value because it will cause the least amount of change in the value as discussed above) by one of bits of the binary that makes up 149. The modified binary form of the pixels of the cover image having the secret data embedded in it is shown below:

Binary code of modified carrier data: 10000101 10000110 10001000 10001101 01111000 01100101 01001010 00100111. These new binary values change the values of the image file very little, with a difference of only one in either direction. These discrepancies are negligible, as humans can't tell the difference at such small levels.
So message is hidden in the image file. The LSB of the modified image file are extracted to obtain the embedded message. For example in the above binary sequence the underlined bits are extracted to obtain the secret message. The binary sequence extracted from above sequence is "10010101".This was the information which was embedded. So the secret message is extracted from the modified or the stego image.



Data hiding inside a image.

ii) Audio Steganography (Hiding Data in Audio Files)

Encoding secret messages in audio is the most challenging technique to use when dealing with Steganography. This is because the human auditory system (HAS) has such a dynamic range that it can listen over. To put this in perspective, the (HAS) perceives over arrange of power greater than one million to one and a range of frequencies greater than one thousand to one making it extremely hard to add or remove data from the original data structure. The only weakness in the (HAS) comes at trying to differentiate sounds (loud sounds drown out quiet sounds) and this is what must be exploited to encode secret messages in audio without being detected. There are two concepts to consider before choosing an encoding technique for audio. They are the digital format of the audio and the transmission medium of the audio. There are three main digital audio formats typically in use. They are Sample Quantization, Temporal Sampling Rate and Perceptual Sampling. Sample Quantization which is a 16-bit linear sampling architecture used by popular audio formats such as (.WAV and. AIFF). Temporal Sampling Rate uses selectable frequencies (in the KHz) to sample the audio. Generally, the higher the sampling rate is, the higher the usable data space gets. The last audio format is Perceptual Sampling. This format changes the statistics of the audio drastically by encoding only the parts the listener perceives, thus maintaining the sound but changing the signal. This format is used by the most popular digital audio on the Internet today in ISO MPEG (MP3) the more popular encoding methods for hiding data inside of audio.

Bender [6] identifies four possible transmission environments .These environments are:

I) Digital end-to-end environment
ii) Increased/decreased resembling environment
iii) Analog transmission and resembling
iv) "Over the air" environment

In a computer-based audio steganography system, secret messages are embedded in digital sound [5]. The secret message is embedded by slightly altering the binary sequence of a sound file. Existing audio steganography software can embed messages in WAV, AU, and even MP3 sound files. The audio signals are digitized and are then stored in the binary form. Audio files use either 8 or 16 bit values to store the sounds files. 8 bit files allow sounds values to range between 0 and 255 and the 16 bit files allow values from 0 to 65535. By changing the values slightly, we can store our data.

For example, we had an 8 bit audio file with the following values: Decimal form of the audio data: 133 135 136 140 120 100 75 39 In binary, those values would be represented as: Binary form of Audio data: 10000101 10000111 10001000 10001100 0111000 01100100 01001011 00100111 Binary Code of Secret message: 11101101 Now we wanted to hide the binary file 11101101 (237) inside the carrier data. We simply replace the least significant bit in each value (the last value because it will cause the least amount of change in the value as discussed above) by one of pieces of the binary that makes up 213. The modified sequence of binary form of audio file when the secret data 237 is embedded in it is shown below: Binary form of modified carrier data: 10000101 10000111 10001001 10001100 01111001 01100101 01001010 00100111. These new binary values change the values of the audio file very little, with a difference of only one in either direction. These discrepancies are negligible, as humans can't tell the difference at such small levels.

So message is hidden in the audio file. The LSB of the modified image file are extracted to obtain the embedded message. For example in the above binary sequence the underlined bits are extracted to obtain the secret message. The binary sequence extracted from above sequence is "11101101".This was the information which was embedded .So secret message is retrieved back from the modified audio file.

ii) VIDEO STEGANOGRAPHY(Hiding Data in Video)

The video file is basically the combination of the image and the audio. In video there are sequences of the frame which are shown at such a speed that the object appears as moving. These frames are stored in the computer in the binary form. All the information of the frames is stored in the form of 0's and 1's.So in steganography the LSB of the bytes of frame is replaced by the bits of the secret message.

Difference between Steganography and Cryptography

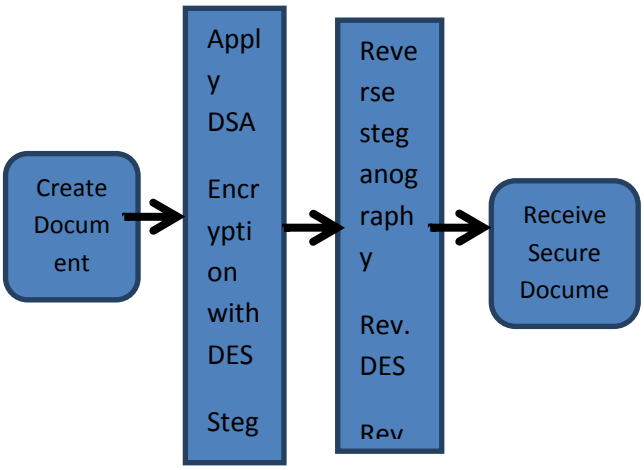
The art of hiding messages is an ancient one, known as steganography. Steganography is the dark cousin of

cryptography, the use of codes. While cryptography provides privacy, steganography is intended to provide secrecy. Privacy is what you need when you use your credit card on the Internet, you don't want your number revealed to the public. For this, you use cryptography, and send a coded pile of gibberish that only the web site can decipher. Though your code may be unbreakable, any hacker can look and see you've sent a message. For true secrecy, you don't want anyone to know you're sending a message at all.

The advantage of steganography, over cryptography alone, is that messages do not attract attention to themselves. Plainly visible encrypted messages no matter how unbreakable will arouse suspicion, and may in themselves be incriminating in countries where encryption is illegal.[1] Therefore, whereas cryptography protects the contents of a message, steganography can be said to protect both messages and communicating parties

III. SOLUTION OF THE PROBLEM

Triple security structure



-architecture of proposed work

This is the architecture for highly secured data by using some security features to enhance the security while transferring data from sender to receiver.
The disadvantage could be the possible fraud by some merchants, also hacking into the electronic records or interception of a transmission is another risk. There is also the danger of human error or equipment failure which can jeopardize the accuracy of transmissions or records. Customers should check their banking records carefully for unfamiliar or unauthorized transactions. So documents are not much secure until unless some security does not provided to it so as the solution to the problem we provide "data with high security" by using some security concepts:-

DSA (Digital Signature Algorithm):-Electronic Signature can prove the Authenticity of Alice as a sender of the message.

DES (Digital Encryption Standard):-DES was designed by IBM and adopted by the U.S.govt.as the standard encryption method.

Steganography: - Steganography is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message.

IV. METHODOLOGY

In this architecture, sender will create a document & then transfer this document to the receiver as a plaintext. If any user wants to transferring data to the intended receiver then its security becomes compulsory so we can apply security algorithms i.e. DSA (Digital Signature Algorithm):-Electronic Signature can prove the Authenticity of Alice as a sender of the message. DES(Digital Encryption Standard):-DES was designed by IBM and adopted by the U.S.govt.as the standard encryption method.

Steganography: - Steganography is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message. The goal of steganography is to avoid drawing suspicion to the existence of a hidden message. This approach of information hiding technique has recently become important in a number of application areas. Digital audio, video, and pictures are increasingly furnished with distinguishing but imperceptible marks, which may contain a hiding copyright notice or serial number or even help to prevent unauthorized copying directly

After applying security algorithms plaintext is converted into cipher text(encrypted message).Then user again apply reverse techniques to get the original message. Then encrypted message again converted into plaintext (original message) in secured manner.

V. IMPLEMENTATION

The performance is based on a min-priority queue implemented by a Fibonacci heap and running in $O(|E| + |V| \log |V|)$

An upper bound of the running time of this algorithm on a graph with edges E and vertices V can be expressed as a function of $|E|$ and $|V|$ using big-O notation.

For any implementation of vertex set Q the running time is in $O(|E| \cdot dk_Q + |V| \cdot em_Q)$, where dk_Q and em_Q are times needed to perform decrease key and extract minimum operations in set Q, respectively.

The simplest implementation of the this algorithm stores vertices of set Q in an ordinary linked list or array, and extract minimum from Q is simply a linear search through all vertices in Q. In this case, the running time is $O(|E| + |V|^2) = O(|V|^2)$.

For sparse graphs, that is, graphs with far fewer than $O(|V|^2)$ edges, this algorithm can be implemented more efficiently by storing the graph in the form of adjacency lists and using a binary heap, pairing heap, or Fibonacci heap as a priority queue to implement extracting minimum efficiently. With a binary heap, the algorithm requires $\Theta((|E| + |V|) \log |V|)$ time (which is dominated by $\Theta(|E| log |V|)$, assuming the graph is connected).

VI. CONCLUSION

By using these three algorithm combination we can secure our communication on the network.By using the combination of these three algorithm we can implement security in the network. Because digital signature algorithm provide signature which means that document is authenticated by the sender. Data encryption algorithm provide encrypted data so that data is save from outsider attacks.and at the receiver end we can decrypt that data. Stegnography can change presentation of data so that nobody can understand that secret data.Thus we can provide security to our data on the network.

REFERENCES

[1]. An Efficient Implementation Of The Digital Signature Algorithm P. Kitsos, N. Sklavos And O. Koufopavlou Vlsi Design Laboratory Electrical and Computer Engineering Department University of Patras. Patras, GREECE E-mail: pkitsos@ee.upatras.gr

[2]. An Introduction to Cryptography and Digital Signatures Author: Ian Curry March 2001 Version 2.0M.M. Amin, .M. Salleh, S. Ibrahim, M.R Katmin (2003), "Information Hiding Using Steganography", 4th National Conference on Telecommunication Technology Proceeding 2003 (NCTT2003), Concorde Hotel, Shah Alam, Selangor, 14-15 January 2003.

[3]. Nameer N. EL-Emam, Hiding a Large Amount of Data with High Security Using Steganography Algorithm Applied Computer Science Department, Faculty of Information Technology, Philadelphia University, Jordan

[4]. Bellare,M., Miner,S.: A Forward-Secure Digital Signature Scheme. In: Wiener, M. (eds.):

[5]. National Institute of Standards and Technology (NIST), Digital Signature Standard, FIPS PUB 186-2, http://csrc.nist.gov/publications/fips/fips186-2/fips186-2.pdf

[6]. Abdalla,M., Reyzin,L.: A New Forward-Secure Digital Signature Scheme. In: ASIACRYPT 2000,LNCS, Vol.1976, pp. 116-129. Springer-Verlag,(2000).

[7]. Anderson,R.: Invited Lecture, Fourth Annual Conference on Computer and Communications Security,ACM, (1997).

[8]. T Morkel, JHP Eloff " Encryption Techniques: A TIMELINE APPROACH" published in Information and Computer Security Architecture (ICSA) Research Group proceeding.

[9]. Text book William Stallings, Data and Computer Communications, 6eWilliam 6e 2005.

[10]. Md. Nazrul Islam, Md. Monir Hossain Mia, Muhammad F. I. Chowdhury, M.A. Matin "Effect of Security Increment to Symmetric Data Encryption through AES Methodology" Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing 978-0-7695-3263-9/08 DOI 10.1109/SNPD.2008.101 IEEE 2008.

[11]. Joan Daemen and Vincent Rijmen, AES submission document on Rijndael, June 1998.

AUTHORS PROFILE

Richa Arya is a M.Tech. scholar pursuing M.tech. from NGF college of engineering and technology Palwal(India).

RICHA
380/A,New Colony Extn.
Rasulpur Road
PALWAL
HARYANA, INDIA
Mob : 9812028933
E-Mail Id: - ricsarya@gmail.com