Use of Steganography in Cloud Computing

Abstract

In today's age of digitalization where more and more individuals as well as businesses store their sensitive data in cloud systems, it is important to ensure that these cloud systems are secure. However, the information technology sector has been witnessing a constant rise in data breaches and cyber-attacks. Though there has been tremendous advancement in cloud services, the industry is still faced with some pressing security problems. Some of these issues include source location, multi-tenancy problems, validation of data obtained and its dependence on the owner, system observing, etc. This calls for implementation of strategies that can enhance cloud security. One such popular technique is steganography. In this research, four steganography techniques (LSB, DCT, PVD, and SST) have been compared with one another on the basis of time taken, CPU utilisation, and image compression in order to determine the most effective technique that helps in enhancing cloud security. It was found the LSB has one of the fastest algorithms, low CPU utilisation, and least image compression, making it an ideal steganography technique.

Keywords: *Cloud computing, Cloud security, Steganography, Least Significant Bit, Discrete Cosine Platform, Pixel Value Differencing, Spread Spectrum Technique.*

Introduction

Steganography

Define

In the modern era, the Internet has become an important source for securing data in the IT sector. To assure that the data is well protected, an encryption technique was implemented to keep the data confidential. This technique was known as Cryptography, a method which is used to encrypt and decrypt secret messages. As the content of the private message was secured, it was also required to keep the original message as confidential as the secret message. The method which was implemented for this purpose was named Steganography.

Steganography is a method which secures information by hiding it in a multimedia format (Image, Audio,Video). In simple words, steganography is the art of hiding confidential data. The motivation to use steganography is to make sure the data which is embedded into multimedia should be hidden from both user ends (encryption and decryption) to make it difficult for the

hackers if it is discovered. The word steganography has its roots in the Greek language, “Stegos” meaning hidden/covered/ or roof, and “Graphia” simply meaning writing (Dickman, 2007).

History of steganography

Steganography owes its discovery to a Greek ruler in 440 BC, Histaeus. This was a revolution for Greece where the soldiers had to shave their head, carve the message on their heads, and wait till the growth of hair to ensure the message was hidden. This method involved carving of the head with the help of wax protected wooden slabs. The message was engraved on the wood and a layer of wax was poured to cover it. To receive the message, the receiver had to shave off the soldier’s head in order to decrypt the message. This same procedure was repeated to reply (Silman, 2001).

As time evolved, the technique of steganography brought hope for adapting new methods. During World War II soldiers used to send the message with the help of invisible ink. The message was decrypted only when the receiver had the appropriate chemical compounds in order to reveal the hidden message (Dickman, 2007). In other parts of the world, a French photographer used to send large chunks of messages with the help of pigeons. While Paris was in major attack by the troops in 1870, they used microfilm to send messages via pigeons. This technique led to the invention of microdot. During the world war, microdot was used as a steganographic technique for transmitting messages. In this technique, the message can be in the form of paper or picture which can easily shrink as small as a dot from a pencil. It helped ensure authenticity and security of the message (Siper, Farley and Lombardo, 2005). However, a major drawback of this method was noticed by the countries when it was easy to detect the covered message. A suggested principle was implemented to keep the cover message undetected. It was proposed that a key should be generated between the cover file and original file to make sure the message has not been accessed by the hacker. This key had the functionality of cryptography which maintains the information secured (Lakshmi and Latha, 2013).

Steganography Techniques

What is an image?

Images have been a strong source for data hiding which was used as the basic principle in steganography. In different domains, images can be of different sizes and formats depending on the application. A suitable algorithm can be used as a steganographic technique for images. The image is specified as a graphical representation in the world of computers. These images are arranged in a grid pattern with every grid having a point referred to as “pixel (bits)”. A computer inspects these pixels as an array of numbers formed in a two-dimensional structure which represents different intensities of light across the grid. Each pixel represents binary digits indicating the colour and location arranged in a row by row from left to right to form an image.

Concept

As the evolution of steganography made a mark in the world of information technology, the majority of digital files started using different methods of steganography. However, it is only effective if the digital objects have a high degree of redundancy. Redundancy is defined as the object bits that produce a precise rate than the original digital fill in which the user can see and use. Most of the images and audio files imply redundancy for information hidden. The main key of redundancy is that these bits can be changed without any detection of alteration (Morkel et al., 2005). There are 4 types of steganography which are as follows (Ahsan and Kundur, 2002):

- Text Steganography: A technique which allows to hide information in the form of text. This method involves data encoding of each letter of the hidden information.
- Image Steganography: A technique which uses different images as a cover where every pixel is a key to reveal the hidden information in an image.
- Audio Steganography: This refers to a technique in which a secret message is concealed in an audio signal which is modified according to the binary sequence of the audio file.
- Video Steganography: It can also be called as the combination of image and audio steganography where large amounts of information can be hidden. The data can be embedded in two forms such as embedding the data in raw video and compressing it later or embedding the data directly into the compressed data stream.
- Network Protocol: In this technique the message is hidden within the network. TCP/IP (Transmission Control Protocol/Internet Protocol) header packets can be a great example to store the data during the time of transmission.

Image Steganography

As image steganography is widely used, implementation of the steganography algorithm depends on various cover objects used and various domains employed for embedding or modifying the cover object. Images have a high level of redundancy and can offer increased capacity and resilience to distortion. In image steganography techniques, the hidden text was stored in an image referred to as noise which can be undetectable for Human Visual System (HVS). Data hiding can be a concern because images represented in static form can be subjected to a range of manipulation, which can lead to non linear operations such cropping, blurring, filtering and compression that can impact on the result of loss of data (Bender et al., 1996). For the purpose of this research, image steganography will be the chosen method to assess its effect on cloud computing security.

Domain Structure

The most common and widely used domains for image steganography are as follows:

- Spatial Domain: This domain represents a two dimensional matrix where the secret information is embedded at the pixel level. The manipulation of the image is applied by modifying the pixel value and encrypting the secret message. Since the Human Visual System (HVS) cannot detect the changes of pixels, the domain can be used to hide the message without being noticed. This technique includes operations such as colour values, modifying the pixel location, and adjusting the intensities of the light on the grid. Least Significant Bit (LSB) is the most common technique to encode the message once modification of the pixel is successfully altered.
- Transform domain: This domain presents a mathematical transformation of the image from its original spatial domain to a different domain such as a frequency domain or wavelet domain. Transformation of the image can be performed on the basis of transformed coefficient to apply the secret message. Comparatively, the Wavelet domain can also be applied to decompose the image into different frequency sub-bands. The highest frequency band can be altered to encode the secret data, as it will be undetectable for human vision. As it requires more computational resources and knowledge to extract the data from the modified coefficient, Discrete cosine transform (DCT) and Discrete Fourier transform (DFT) can transform the image into frequency components (Chedda et al., 2010)

Steganography is a process in which information and data is hidden in multimedia files like images, texts, audios, and videos. It embeds secret data using specific algorithms which are then decoded at the end of the receiver.

The following steganography techniques will be employed for the purpose of this research:

1. Least Significant Bit (LSB)

   LSB is the most widely used steganography technique and also the most difficult. It is an insertion technique used in image steganography to hide a secret message within an image file. It works by replacing the least significant bit of the pixel value in the image

with a bit from the secret message. This involves masking, filtering, and transforming. It works on the principle of replacing the Least Significant Bits (LSB) of the cover image with the Most Significant Bits (MSB) of the information that is to be hidden without significantly hampering the properties of the cover image (Raja et al., 2006).

2. Discrete Cosine Transformation (DCT)

   DCT is a steganography technique which implements breaking of image into 8x8 block of pixel. After breaking down the image, a general equation is added to the multidimensional block which executes the equation from left to right and top to bottom. DCT converts the image into a frequency which is divided into regions such as High Frequency (FH), Low Frequency (FL), Middle Frequency (FM).

3. Pixel Value Differencing (PVD)

In this steganography technique, information is hidden within digital images. It is mostly performed in the spatial domain. Essentially, PVD modifies the least significant bit (LSB) of each pixel in the image on the basis of the difference between the current and neighbouring pixel value. After calculating the difference between each pixel and its adjacent pixel, if the value surpasses a predefined threshold value, the LSB of the pixel is transformed to encode a part of the secret message. If the value is lesser than the threshold value, the LSB of the pixel remains unaltered (Fridrich et al., 2001). Though it is a simple and effective technique, it is vulnerable to attacks such as those involving statistical analysis of the data in the image. Additionally, if a large amount of bits are encoded, it may result in noticeable visual artefacts i.e. it has a low embedding capacity (Luo et al., 2011).

4. Spread Spectrum Steganography (SSS)

This steganography technique spreads information over a broad frequency range in order to hide information within digital signals. It uses a spreading code to embed the secret information into the host signal. This host signal can be a digital image, audio, or video. In this technique, first the

secret message is converted into a binary sequence. Then, the sequence is multiplied with a pseudorandom spreading code, resulting in its spread over a wide frequency range. This spread signal is then added to the host signal. The host signal may contain interference in various forms such as noise. The same spreading code which is used to embed the secret message is deployed to despread the signal in order to decode the secret message from the spread signal. To extract the secret message, the despread signal is processed (Marvel et al., 1999).

Research Question

Which steganography technique is the most useful and efficient in reducing attacks on cloud system security?

Research Objectives

1. To conduct a thorough literature review to identify potential research gaps in research revolving around the use of steganography techniques to improve cloud security
2. To determine the weaknesses in the security of cloud computing systems
3. To identify the current trends in steganography techniques that can be used to enhance cloud security
4. To perform a comparative analysis of steganography techniques in order to ascertain which technique is the most effective in preventing attacks on cloud systems

The present research aims to identify the contribution of steganography on security in cloud computing systems. Thereafter, it will determine which steganography technique is the most effective in enhancing the security of cloud computing systems. It also aims to contribute to the field by providing recommendations to overcome the present problems in the area.

Review of literature

Though the innovation of cloud computing systems brought about a revolution in the field integrating IT services and the business sectors, it poses significant security issues from various perspectives including stakeholders, architecture, characteristics, etc. (Almorsy et al., 2016). Cloud Security refers to the risk and vulnerabilities associated with storing and processing the data in the cloud. It is a part of computer security and encompasses a set of technologies, policies, and control methods which assist in protecting the data in the cloud and the services that it provides. The fact that cloud systems provide various benefits like flexible access to data, omnipresence of data, and resilience allows cloud providers to allocate their funds in improving security (Ryan, 2013). But due to the nature of the system itself, when the data is uploaded onto the cloud it is inevitably put out of reach of the clients, leading to the rise of significant security issues. Singh and Chatterjee (2017) identified the following challenges that are currently faced by the cloud industry:
This brings many advantages, including data ubiquity, flexibility of access, and resilience.

1. Problems regarding security of stored data and computing
   Data is central to any cloud computing service. After the data is in the cloud, the clients are unaware of what happens to it, making it isolated and inscrutable to them. This problem is evident in two situations i.e. before the data is computed and after the data is computed.
   - Issues with data storage: A critical issue that occurs is the confidentiality of data stored in the cloud servers by the clients.For example, big companies like Google and Amazon have physical servers located in various countries and they store the data based on a feature called multi-location which causes security and legal problems because different countries have different policies.
   - Un-trusted computing: The overarching aim of many security services is to provide a front end interface for SaaS applications when users request a web service. These applications adapt and change according to the pattern of usage and behaviour displayed by the user. Such computing frameworks may produce unwanted, false, or inaccurate results caused by malicious or misconfigured servers.
   - Availability of data and services: Though cloud resources and services are highly available, both virtual and physical, they require changes to be made at the application as well as infrastructural level. A potential solution to this is multiple alive servers supporting the applications, however, this makes the cloud system vulnerable to Denial-of-Service (DoS) attacks. Such attacks involve flooding the target system or application with information, or sending information triggering an attack, resulting in the user being shut out of their network or machine. Partial or complete failure of the system and availability of services and data may result

from even a single fault in the cloud system environment. Additionally, cloud outages may be the resultant of unavailability of hardware resources. Such problems can cause immense distress to the business availing the cloud services.

- Cryptography: This mechanism is deployed to secure the information stored in cloud systems. It involves converting plain text into cipher text. However, if the algorithm is implemented in an ineffectual manner or uses a weak key during the encryption process, it makes the data susceptible to attacks, for example, brute force attacks.
- Recycling of cloud data: It is essential for the cloud service providers to sanitise the cloud space once data has been utilised and deleted by the user. Improper sanitization can lead to data leaks and losses.
- Malware: Malware has the ability to harm cloud services and devices by deleting or corrupting data stored in the cloud. It spreads to all other files which are synchronised with the file containing the malware.

2. Problems with security of virtualization

   A major factor which has contributed to the widespread adoption of cloud technology in the business industry is the virtualization of services. Though it contributes positively by reducing costs and increasing profit, it is still vulnerable to threats and attacks.

   - Image management in virtual machines: Features like elasticity and service orientation create a dynamic and volatile environment for the users to create, modify, and copy virtual machine (VM) images and even allows for the usage of previously created images. This poses severe security problems, for example, a malicious user has the ability to upload corrupted images in the repository containing malware, or can even find possible attack points by finding the codes of images. Improper management of VM images can result in breach of confidentiality of users, VM sprawls, or wastage of resources.
   - Virtualization of networks: Attackers can access sensitive information of the clients or network providers as virtualization of networks paves the way for limited administrative access to the cloud and network tailoring caused by instability in network characteristics, abnormal packet delays, and unstable OSI layers. It also leads to other security issues like spoofing, packet sniffing, and network based VM attacks.
   - Mobility: The process of copying or moving a VM to other servers is known as VM cloning or template image cloning. This process can lead to multiple errors in security like misconfiguration as the mobility of VM aids for quick development of VM images, hampering the process of transferring and increasing deployment time.

3. Security issues related to internet and services
   Along with services and resources, the cloud infrastructure requires a carrier to transmit data between senders and receivers. This carrier is the internet. In the form of digital data, the internet transmits a large number of data packets from the source to the destination passing through a number of nodes, making it unsafe for the end users.
   - Venomous outsiders and Advanced Repeated Threats (ARTs): ART refers to an attack model with consists of the three phases i.e. information gathering via public or private intelligence sources aka Open Source Intelligence (OSINT) gathering phase, threats modelling phase, and finally, the attack phase. Hence, it is important for businesses to keep essential information private and keep a tab on what type of information and how much of it is provided publicly.
   - Web services: Data integrity is a critical problem in a distributed system. The Service Oriented Architecture (SOA) provides a solution to maintain integrity of data in clouds. Extensible Markup Language (XML) and Application Programming Interface (API) are used to improve functionality of clouds. However, APIs do not provide any transactional reports which can exacerbate data integrity issues.
   - Availability of safe web services: Majority of cloud services are accessed via web based agents like web browsers. There are vulnerabilities like malicious web links and websites which constantly increase the malware in cloud systems, therefore, attracting attackers to attack vectors.

4. Security issues in networks
   Networks are fundamental to cloud computing systems, hence, network level issues have a direct negative impact on cloud environments. Due to the dynamic nature of cloud networks, these issues are considered internal as well as external. DoS attacks on networks do not only affect the availability of the services but also affects the bandwidth of networks and can cause congestion in networks. Network security issues lead to vulnerabilities through mobile platforms and circumference security.

5. Problems with access control
   Protection from unauthorised permissions is called access control security. A combination of an e-mail address or username and password is used to authenticate the identity of users in order to maintain access security in order to prevent attacks via web technologies or websites. This includes physical access, user credentials, authorization, authentication of entities, management of user identities, and anonymization.

6. Problems regarding software security
   A major concern in the present day is that of software security. This is because of the wide array of programming languages available. Cloud computing systems are composed

of thousands and millions of lines of code. Due to this, there is a lack of a single universally accepted system to measure the levels of security in cloud systems. Security issues can arise by a single bug in the code, which can occur at a framework level as well as user interface level.

7. Problems with management of trust
   A relationship built on trust is essential between the client and the cloud service provider. It acts as a basis to access various resources and services such as storage, web based access, visualisation mechanisms, and computational algorithms. Trust is also imperative in peer to peer networks, sensor networks, and distributed networks. Therefore, it is important that the human aspect be emphasised while handling security.

8. Problems regarding compliance and legal issues
   An essential document in the cloud-business model is the Service Level Agreement (SLA). It is signed by both parties i.e. the client and the service provider, and it outlines the services or resources which will be delivered, the terms and conditions of usage, the metrics by which effectiveness of the processes for monitoring, etc.
   - Digital forensics: The recent years have witnessed a boom in the field of digital forensics which are used to audit tasks in cloud computing with an overarching aim of identifying potential threats. Issues arise when resources can not be located and isolated, called data locality issues. Seizing of data and disclosure of data can compromise the privacy and confidentiality of clients.
   - Legal problems and cyber laws: Cyber laws are backdated and also may potentially breach the confidentiality and privacy of users as these cyber laws do not provide complete security to cloud systems. As the physical servers used by cloud systems are located in different countries all around the globe, the laws are not universal and change from country to country. This can lead to breaches in the SLA and nonconformity on the part of providers due to the feature of data migration.

Steganography has been found to be a useful tool by many researchers which enables covert transmission of information through overt communication channels. It combines encryption methods with cryptography techniques which allows for the user to send information hidden inside a multimedia file in plain view. When combined with pre-existing encryption algorithms, the hidden data becomes difficult to detect as well as decipher (Dickman, 2007). Current researches have proven that cloud security can be enhanced using a combination of methods like steganography, encryption and decryption techniques, compression and splitting techniques, digital signature algorithm, data encryption standard algorithm, etc. in order to bypass the limitations of previously used traditional data protection methods (Awadh et al., 2019). It has

also been found that steganography techniques can be employed without the interference of a third party (Ahmed and Abduallah, 2017).

One of the most widely used steganography techniques is Least Significant Bit (LSB). This technique involves encoding a secret message within the least significant bits of a multimedia file in image or audio format. As the least significant bits of a digital file represent the least important information of the file, altering them to encode the message does not have a significant effect on the overall quality of the file. LSB has been used in various areas such as digital forensics, military communications, medical imaging, etc. It was found that LSB has several advantages like it minimises the error rate in the process of embedding the secret message and that it has greater criterion reliability (Rahman et al., 2022). It has a high capacity of carrying information and is a simple method to implement (Goli and Naghsh, 2016). Additionally, the information can be retrieved without any loss of quality of the media file (Ali et al., 2020). However, despite its advantages, it also has certain limitations. LSB was also found to be vulnerable to attacks, for example, compression, cropping, and salt and pepper noise (Goli and Naghsh, 2016). Another challenge is that the storage capacity of the LSBs may not be enough to hide large amounts of confidential information. Unauthorised users may also be able to detect secret messages which jeopardises the confidentiality of the data (Cheddad et al., 2010).

Another popular steganography technique is Discrete Cosine Transform (DCT) which works on the principle of modifying the coefficients of an image's frequency domain in order to embed secret data. The least significant coefficients are modified to ensure that data is hidden while ensuring minimal damage to the quality of the digital file. It also involves applying a secret key in the process of encoding and decoding the secret message from the image. This key controls the positions of the coefficients that have been altered (Patel and Dave, 2012). Over the years, studies have found DCT to be an effective and reliable steganography technique which can be used to transmit information confidentially. A study in which DCT was implemented in different image formats like JPEG, PNG, and BMP, it was found that the technique is effective in embedding secret messages while maintaining the integrity of the original images (Zhang et al., 2019). Another study analysed DCT on the basis of data hiding capacity, image fidelity, and security. The results indicated that DCT was more effective than other steganography techniques (Wang et al., 2017). At the same time, DCT has its limitations. It was found that DCT steganography is susceptible to attacks like histogram analyses which aids in the detection of hidden information. Also, DCT is a time taking and computationally intensive procedure (Zhu et al., 2021).

Another established steganography technique is Pixel Value Differentiation which involves modifying the pixels surrounding those pixels holding the least value based on the difference of values between the pixels in order to embed secret messages. A secret key, which controls the positions of the pixels that have been altered, is required in order to encode and decode the

message. The modification is done in a manner which is undetectable to the human eye. Studies have shown that PVD is preferred over LSB as the process of embedding is smoother in PVD steganography. A major advantage of this technique is the imperceptibility of the modification of the image. However, at the same time, this method lacks security, for example, it is susceptible to histogram analyses which can lead to the detection of secret messages in the altered images (Kadhim et al., 2019). To overcome these weaknesses, modified versions of PVD were also introduced such as Adaptive PVD, a combination of PVD and LSB, improved rightmost digit replacement (iRMDR) and parity-bit pixel value difference (PBPVD), etc. (Hussain et al., 2017).

Coming to the last well known steganography technique known as Spread Spectrum Technique. It is a technique which involves the data to be hidden within a digital signal. This technique makes it difficult to detect the secret information. An application was deployed where the technique generates a key to encrypt the secret message before embedding it in the image. The advantage of the key is to decode the message from the image . The system proposed by the research was to divide the message into small blocks. These blocks use a pseudo random sequence to select specific blocks to embed the secret data . The messages are spread all over the signal so that the authorised users have access to decode the information. Digital signature can be great example authentication for verifying the integrity of the watermarked image and copyright protection (Zhelezov and Kordov, 2020).

Steganography has been found to be the most effective technique to enhance the security of cloud computing systems. This is because it makes it difficult for attackers to breach confidentiality and compromise the integrity and authenticity of data. Discussed below are the characteristics of steganography techniques which aid in improving cloud security (AlKhamese et al., 2019; Ahmed and Abdullah, 2017):

1. Confidentiality - It can prevent unauthorised access to attackers by embedding confidential information in multimedia files, hence, concealing sensitive data.
2. Integrity - Checksums or digital signatures can be integrated within data files which maintain the integrity of data stored in the cloud systems. By doing so, cloud providers can ensure that the data has not been tampered with in the process of transmission or storage.
3. Authentication - Cloud service providers can also use steganography techniques to demonstrate authenticity and veracity of the data being stored in their system. They can also detect any alterations. This is done by embedding digital signatures like watermarks into the files.
4. Detecting unauthorised access - Steganography techniques can be employed to embed “triggers' ' in the form of data in files that alert the administration of cloud service providers if the files are accessed without authorisation and authentication. This enables cloud service providers to take immediate preventive or corrective actions in case of a security breach.

However, there is ambiguity over the efficacy of different techniques of steganography and their usage in different scenarios relating to cloud computing. Furthermore, present literature outlines various limitations like the fact that current techniques and schemes are only able to tackle a limited number of security issues which arise in cloud computing environments and that too in a small environment. The aim of this research is to identify the most effective steganography technique which can be used to enhance cloud computing security. Additionally, recommendations will be provided on the basis of the findings in order to bring improvement in steganography and encryption-decryption techniques on a large scale to solve multiple security problems in a wider environment.

Methodology

3.1 Summary of the application:

- **Understanding of basic Steganography model**

  With the rise in information security risks, image steganography has become an important technique for data transmission as it guarantees the secrecy, security, and confidentiality of the information. Explained below is how steganography simply works:

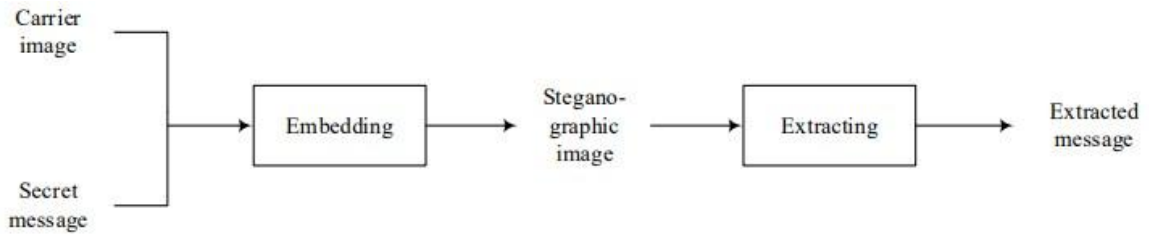


Figure figure number?: Basic Steganography model

There are two major steps which ensure the confidentiality of the information. They are as follows:

- **Embedding:** The process of hiding a secret message into the carrier image.
- **Extracting:** The process of accessing the information from the encoded steganographic image

- **Application**

After a thorough analysis of the present literature on the different approaches for image steganography, an application was developed to understand and analyse which steganography technique is most suitable for data hiding. In this study, four methods have been implemented to assess the efficiency of the steganography technique with respect to data security. They are as follows:

- **Least Significant Bit (LSB):** This algorithm is categorised into two scripts i.e. LSB_Encoding and LSB_Dedcoding.
  1. **LSB_Encoding:** This script defines a function called "encode_image" which takes an image and a secret message as an input parameter. The function encodes the secret message into the image by converting the messages into bytes, checking the size of the messages, converting the message from bytes to binary, and embedding the binary message into the image. Once all the conditions are satisfied the function returns the encoded image.

The algorithm requires importing of necessary python libraries which provides access for the functions. These libraries are hosted by the python interpreter. They are as follows:

- **SYS:** This module is used to interact with the python environment. It grants access to various variables and functions to maintain and manage the arguments accessed through the command line by the interpreter.
- **PSUTIL:** Process and system utility is a cross platform library which is used for monitoring the system and utilisation of the resource. For the purpose of this research, the application retrieved information on the running processes and gathered information such as CPU utilisation, time taken to encode the message, and file size after and before compression.
- **PIL:** Python Image Library enables the interpreter to deal with images.

Once all the necessary libraries had been imported, "time" and "image" modules were imported from the python image library. The "time" provides functions related to time and functions to analyse the time taken to encode the secret message. Once the secret message is embedded into the carrier image, the image converts into RGB colour space by implementing the "convert" method. It converts the steganography file and stores it in ".jpeg" format in the same directory where the encoded message is saved. Finally, the algorithm calculates the parameters in order to analyse this technique.

2. **LSB_Decoding:** This script describes a function called "decode_image" which fetches the file path of the image and returns the hidden message encoded in the image. This function opens the encoded image and converts into an RGB format. Image function extracts the least significant bit of each pixel to reconstruct the binary message after extraction of the image into a binary format. Character Detection which is also known as "chardet()", this package helps to detect the hidden text and extracts the encoded message from the steganography image.

- **Spread Spectrum Technique:** This algorithm is categorised into two scripts i.e. SST_Encoding and SST_Dedcoding.
    1. **SST_Endcoding:** A function known as "embed_sst" is implemented which plays an important role in encoding procedures. This function is used to find the image path, the secret message, and embeds two parameters i.e. "alpha" and "beta". This function was implemented to find the strength of embedding the secret message into the carrier image. The code uses a function "str2bits" which enables a string message as input and returns a binary string message of 0's and 1's. The strength was calculated by a threshold value based on the "alpha" parameter.

"Embed_coeffs" is an array which was implemented for storing the strength values. Initially, the value was set to zero using the "np.zeros_like" function which sorts the values in an ascending order using the "np.sort" function and compares it with a formula which involves the parameter of beta. After calculating the strength, it embeds the secret message into the cover image using a technique called spread spectrum steganography. Libraries implemented for the working of this technique are as follow:

- "**Sys"** was imported to maintain and manage variables and methods for the function "embedd_sst".
- "**Psutil"** was imported to measure and monitor functions such as CPU utilisation, processing time, etc.
- "**Numpy"** was imported to perform the numerical operations on array.
- **"Scipy.fftpack"** was imported to perform an important mathematical expression of fourier transformation.
- "**PIL.image"** was imported to handle complex procedures of image processing.

Thus, after the completion of implementing the function and libraries, the stego image was stored in a local database labelled as "encoded.jpeg". A report was also generated on the basis of CPU utilisation, file compression, and time taken.

2. **SST_Decoding:** A function was implemented known as "extract_sst" which passes three parameters : "stego_image_path", "alpha", "beta". Alpha is an integer representing the number of DCT coefficients which was used for embedding messages. Beta is a scaling factor to detect the embedding strength of the array. DCT computes the images in rows and columns by implementing "T" which is known as "threshold" to transpose the array. In this study, the secret message was extracted from the image by iterating over 8X8 clocks of the "embed_coeffs" array. The messages were computed and examined in such a way that the bits were linked with the binary string named "secret_msg_bin". The binary string message was converted using the "char" function to store the secret message. After the completion of the function, the message was extracted from the stego image. The extraction of the message was printed on the basis of alpha and beta where the value was set to "1000" and "0.01" with respect to the image path.

- **Discrete Cosine Transform (DCT):** This algorithm is categorised into two scripts i.e. Endcode_DCT and Decode_DCT.
    1. **Encode_DCT:** This technique uses two arguments; the first argument is the path file for the cover image, and the second argument is the secret message encoding. In this study, "cv2.dct()" function was implemented to perform the arguments.

The secret message was converted to binary using ASCII encoding. Meanwhile, the carrier image was converted to grayscale image. The function was implemented using 8X8 block grid structure. The image was allocated a block by calculating the height, width of the image. Thereafter, the secret message was embedded in a sequential manner. To fetch the steganography image, inverse DCT function was applied to the modified DCT using "cv2.idct()" function. Predefined libraries were accessed from the python interpreter. They are as follow:

- **SYS:** This library grants permission to maintain the variable and performance of the function "cv2.dct()"
- **CV2:** It is known as the OpenCV library which performs computer vision tasks and aids in image processing.
- **PSUTIL:** This is a hybrid library which retrieves the data from the process and system.
- **Time:** Provides time related functions.
- **Numpy:** It is a predefined library which solves large and complex matrices and arrays with the help of high level mathematical functions.

2. **Decode_DCT:** For retrieving the steganographic image, the image was formatted to grayscale by using the "cv2.cvtColor()" function. The converted image performs a DCT function where the image were converted from frequency domain to spatial domain by passing the "cv2.dct()" function. The secret message was retrieved by analysing each block on the basis of the height and width of the block using array slicing. After converting the messages into bits, "str()" was implemented to convert the bits into binary. Hence, the "chr()" function was used to extract the secret message from the binary message. The libraries that were used for the "encode_dct" step were used for this step as well.

- **Pixel Value Differencing (PVD):** This algorithm is categorised into two scripts i.e Encode_PVD and Decode_PVD.
    1. **Encode_pvd:** This script performs a function known as "encode" where two arguments are implemented; the path of an image file and a secret message to be hidden within the image. "PIL" Python Image library was used to load the image and obtain the dimension of the image with respect to height and width. For embedding the secret message, Format() function was implemented to convert it into a binary format which was stored in a variable named "binary message". This function checks the length of the message and employs a parameter which checks the length of the binary message and pixels of the image. Meanwhile, it performs

an if-else statement which checks each block from top to bottom positioning with respect to (0,0). If the condition is found to be true, the message will be encoded. If the condition executes a large value as compared to the pixel value, it passes a ValueError which does not allow the message to be encoded. The modified image is then stored with the original filename_encoded.jpeg. Predefined libraries were accessed from the python interpreter. They are as follow:

- **SYS:** This library was imported to maintain and manage variables and methods for the function "encode_pvd".
- **CV2:** It is also known as the OpenCV library which performs computer vision tasks and image processing.
- **Numpy:** It is a predefined library which solves large and complex matrices and arrays with the help of high level mathematical functions.
- **Pywt:** It is also known as PyWavelets which is a mathematical technique that allows signals and images to be broken into different frequency components to support multi-level reconstruction of images.

2. **Decode_pvd:** The predefined libraries used in the "encode_pvd" step were used in this step as well. "Extract_text_from_image" function was used which hides text on the basis of high frequency bands. These bands contain detailed information about the image and the message. This function was performed in a loop to extract the least significant bit (LSB) from a "binary_text". The loop terminates only if the eighth consecutive LSB is equal to 1. As a part of extraction, "chr()" function was implemented to convert the binary_text to ASCII characters. In the end, the functions return the embedded secret message.

Basis of analysis

The identification of the most suitable technique was analysed on the basis of the following technical parameters:

- **CPU Utilisation**: Central Processing Unit Utilisation refers to the amount of resources utilised to execute a task or processing of the data.
- **Time**: It refers to the amount of time taken to encode and decode the message from the image.
- **Image Compression**: Image Compression refers to redundancy of the size of image and protecting the essential information to be hidden inside the image.

Research Design/Implementation

Once the user downloads the image received via email, they can upload it to the appropriate page and click on 'Decode.' If the user and the image are valid, the password will be displayed in text format, which can then be used to log in again.

The additional security layer provided by MFA is challenging for cyber attackers to gain unauthorised access to the data even if they have the users credentials. It is the easiest way to implement this security layer where the user only requires to provide some additional authorization layers. It resolves major attacks such as:

- Brute-Force Attacks: Auto Generated tools are accessed by the attackers to try a vast variety of username and password combinations to gain an unauthorised entry to an account. MFA can fight such an attacker as it would still require a second authentication factor to access the account.
- Phishing Attacks: Attackers use fraud websets or spam emails to manipulate the user into giving the credentials to access the account.
- Man-in-the-middle: Intercepting the communication between the user and the cloud system can be a possibility to leak the credentials
- Data & Security attacks : A biometric factor like fingerprint generated like fingerprint generated from an app can be a task for the attacker to access or steal the stored data.
- Virtualization Attacks: These threats are usually inside the infrastructure of a virtual machine. To ensure the authenticity a one time code can be generated as a layer to protect the privileges and resources accessed by the virtual machine. Even if the unauthorised users try to abuse the resources, MFA will alert for an authentication code to grant entry.

Research outcomes and Results

After performing a thorough analysis of all the four steganography techniques i.e. Least Significant Bit, Discrete Cosine Platform, Pixel Value Differencing, and Spread Spectrum Technique, it can be said that each technique has its pros and cons. For example, if one technique has a faster processing time, it will have a larger CPU utilisation making the computation more expensive than other techniques. This is because more CPU utilisation means more resources in use, resulting in an increase in cost.

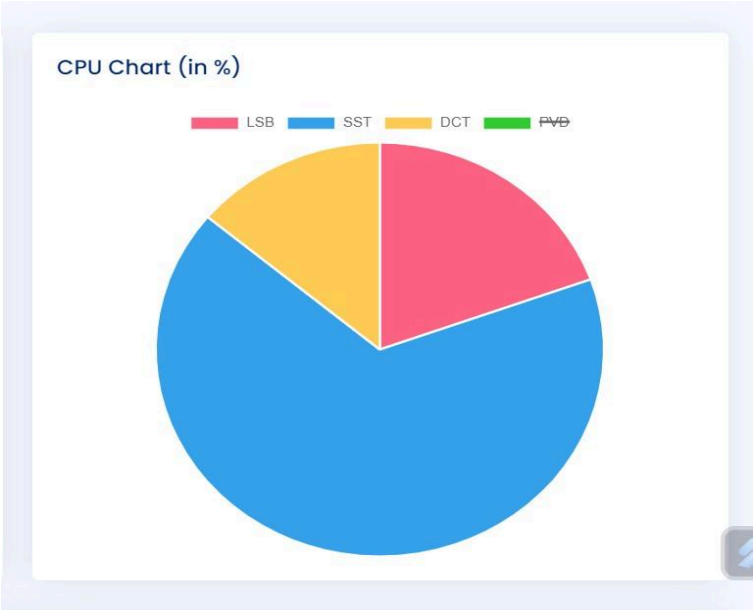
On the basis of CPU utilisation, processing time, and compression of the image, the following results were found. These have been depicted in the form of pie charts for ease of comparison.

Previous executed records

Entries of current user only

#	Time Taken	CPU Utilised	Orginal Image Size(kb)	Encoded Image Size(kb)
20	0.04144692420959473	14.6	34.63	29.12
18	0.04759407043457031	19.8	47.8349609375	125.28
17	0.12690997123718262	23.7	242.4638671875	182.24
16	0.13691997528076172	16.0	242.4638671875	182.24
15	0.13929295539855957	28.3	242.4638671875	182.23
13	0.04263782501220703	23.4	222	129

According to the results of the analysis of the time taken parameter, DCT is the fastest technique with a time taken of 0.023 seconds, followed by LSB with 0.071 seconds, then SST with 0.092 seconds, and lastly, PVD with 0.093 seconds.



According to the analysis of the parameter of CPU utilisation, PVD has the least CPU utilisation with 11.85%, followed by DCT with 12.85%, then LSB with 16.58%, and lastly 56.6%.

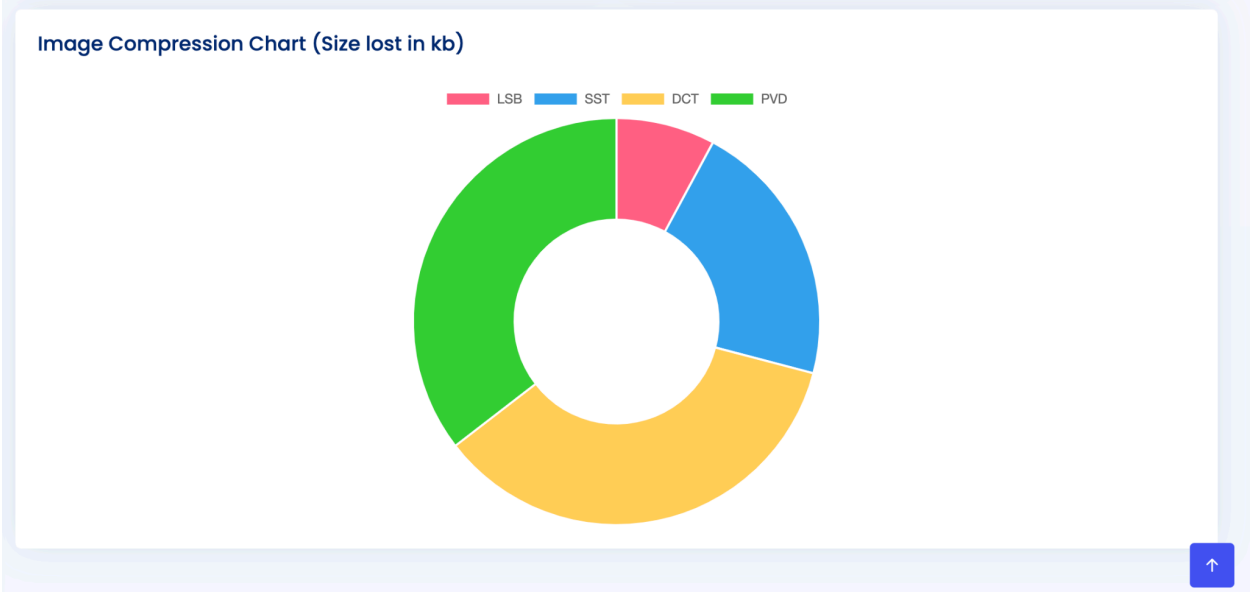


Figure :

According to the analysis of the parameter of image compression, LSB is the most effective technique at 44.954 bytes, followed by SST with 83.759 bytes, then DCT with 139.701 bytes, and lastly PVD with 139.819 bytes.

Overall, LSB appears to be one of the fastest algorithms, making it a reasonable choice for faster computing needs. However, it may not be the most efficient algorithm. Despite this, it still has a relatively small footprint on the pie chart, indicating low CPU usage and potential cost efficiency. One significant advantage of using LSB is that it encodes the image with the least amount of compression.

SST, on the other hand, requires the most CPU resources, making it a costly choice. However, the image compression is not significant.

DCT is the fastest algorithm when it comes to encoding images, but it also results in the most image loss. Similarly, PVD results in similar image compression as DCT. Nonetheless, the significant advantage of using DCT is its speed, which is faster than the other algorithms

Research outcomes/ Discussion

The objective of the application is to evaluate the efficacy of each encoding technique and determine the most suitable technique for the intended purpose. After a thorough analysis of the pros and cons of each technique, the appropriate method for the requirement can be selected.

If the top priority is to minimise image size loss, then the LSB technique appears to be the optimal choice. LSB results in minimal image compression and preserves image quality close to the original image. Additionally, it proves to be a cost-effective and fast technique as it utilises considerably fewer CPU resources and encodes images at an acceptable speed.

After reaching a conclusion, it is practical to apply this image steganography technique to extract real-world usage from it. There are several ways to integrate image steganography into a system to enhance security. Since the research is limited to cloud usage, a potential approach is to implement this technique to strengthen cloud security.

.

Conclusion/Future work

Conclusion

Image Stegnography is considered the most secure technique compared to other steganography techniques. In the continuous advancement of the internet, larger availability of images has been easily accessed and considered a secure way of communicating as it will be difficult to detect where the information is hidden. Although, to ensure the security of the image steganography, using the right technique will guarantee the secrecy of the information.

References

Ahmed, O.M. and Abduallah, W.M., 2017. A review on recent steganography techniques in cloud computing. *Academic Journal of Nawroz University*, *6*(3), pp.106-111.

Ahsan, K. and Kundur, D., 2002, December. Practical data hiding in TCP/IP. In *Proc. Workshop on Multimedia Security at ACM Multimedia* (Vol. 2, No. 7, pp. 1-8). New York: ACM Press.

AlKhamese, A.Y., Shabana, W.R. and Hanafy, I.M., 2019, February. Data security in cloud computing using steganography: a review. In *2019 International Conference on Innovative Trends in Computer Engineering (ITCE)* (pp. 549-558). IEEE.

Almorsy, M., Grundy, J. and Müller, I., 2016. An analysis of the cloud computing security problem. *arXiv preprint arXiv:1609.01107*.

Alshwaier, A., Youssef, A. and Emam, A., 2012. A new trend for e-learning in KSA using educational clouds. *Advanced Computing*, *3*(1), p.81.

Arora, R. and Bajaj, K.S., 2013. Highly Effective Advanced Technology" HEAT" Re-defining Technology for Hospital Management. *International Journal of Management & Behavioural Sciences, Special Edition*, pp.68-73.

Awadh, W.A., Hashim, A.S. and Hamoud, A., 2019. A review of various steganography techniques in cloud computing. *University of Thi-Qar Journal of Science*, *7*(1), pp.113-119.

Bender, W., Gruhl, D., Morimoto, N. and Lu, A., 1996. Techniques for data hiding. *IBM systems journal*, *35*(3.4), pp.313-336.

Bohn, R.B., Lee, C.A. and Michel, M., 2020. The NIST cloud federation reference architecture.