“Security in Ad-hoc network through Image Steganography”

A project report

Submitted in fulfilments for the requirement for the award of the degree of

Bachelor of Technology

In

Computer Science and Engineering

“BEC820 - Project Phase – II”

Submitted By:

Amit Kumar
(U1951038)

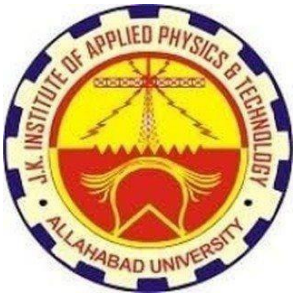
And

Ganesh Prasad
(U1951007)

Under the supervision of

Dr. Vinod Kumar

(Assistant Professor)



Department of Electronics and Communication
Faculty of Science, University of Allahabad
Prayagraj – 211002 (INDIA)

Department of Electronics and Communication
(JK Institute of Applied Physics and Technology)
University of Allahabad, Prayagraj – 211 002

Declaration

This is to certify that the project work entitled "Security in Ad hoc network through Image Steganography" is a bonafide work carried out by *Amit Kumar* and *Ganesh Prasad*, student of B.Tech. (*CSE*), 8th Semester in the Department of Electronics and Communication, University of Allahabad, Prayagraj (INDIA), under the esteemed supervision of "*Vinod Kumar sir*". I declare that the work presented here is carried out by me and has not been submitted anywhere else for the award of any degree/certificates.

Signature

Amit kumar

Signature

Ganesh Prasad

The work presented in this report has been done under my supervision.

Signature
Vinod Kumar sir

INDEX

ACKNOWLEDGEMENT

Although we have taken efforts in this project, it would not have been possible without the kind support and help of many individuals. We would like to extend my sincere thanks to all of them.

We are highly indebted to Vinod Kumar, my project supervisor, for his guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

Our thanks and appreciation also goes to my colleagues in developing the project and people who have willingly helped me out with their abilities.

We would like to thank all faculty members and staff of the Department of Computer Science and Engineering, J.K Institute of Applied Physics & Technology for their generous help in various ways for the completion of this project.

We would like to express our heartfelt thanks to our beloved parents for their blessings, and my friends and classmates for their help, cooperation, and encouragement which helped us in the completion of this project.

ABSTRACT

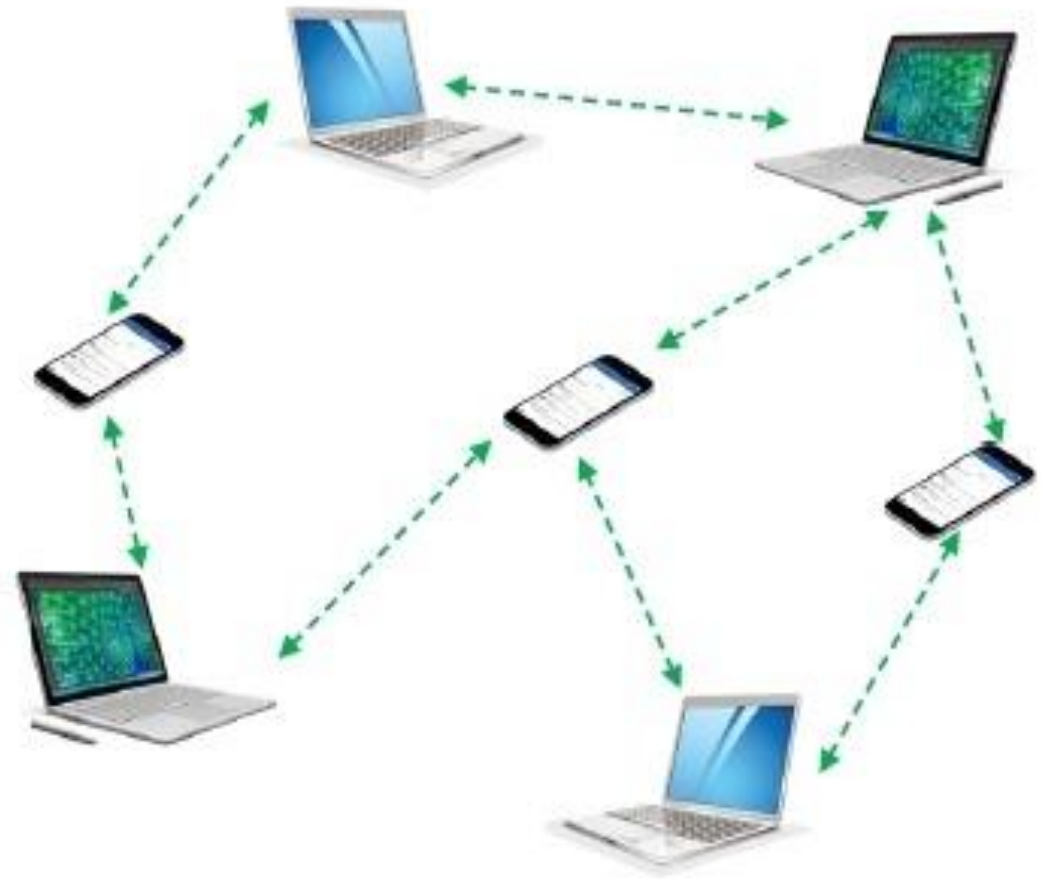
The project deals with learning about the Ad-hoc network and various types of steganography available. Image steganography is performed for images and the concerning data is also decrypted to retrieve the message image. Since this can be done in several ways, image steganography is studied and one of the methods is used to demonstrate it.

Image steganography refers to hiding information i.e. text, images or audio files in another image or video files. The current project aims to use steganography for an image with another image using spatial domain technique. This hidden information can be retrieved only through proper decoding technique. This encryption and decryption of the images is done using MATLAB codes.

1. INTRODUCTION TO AD HOC NETWORK

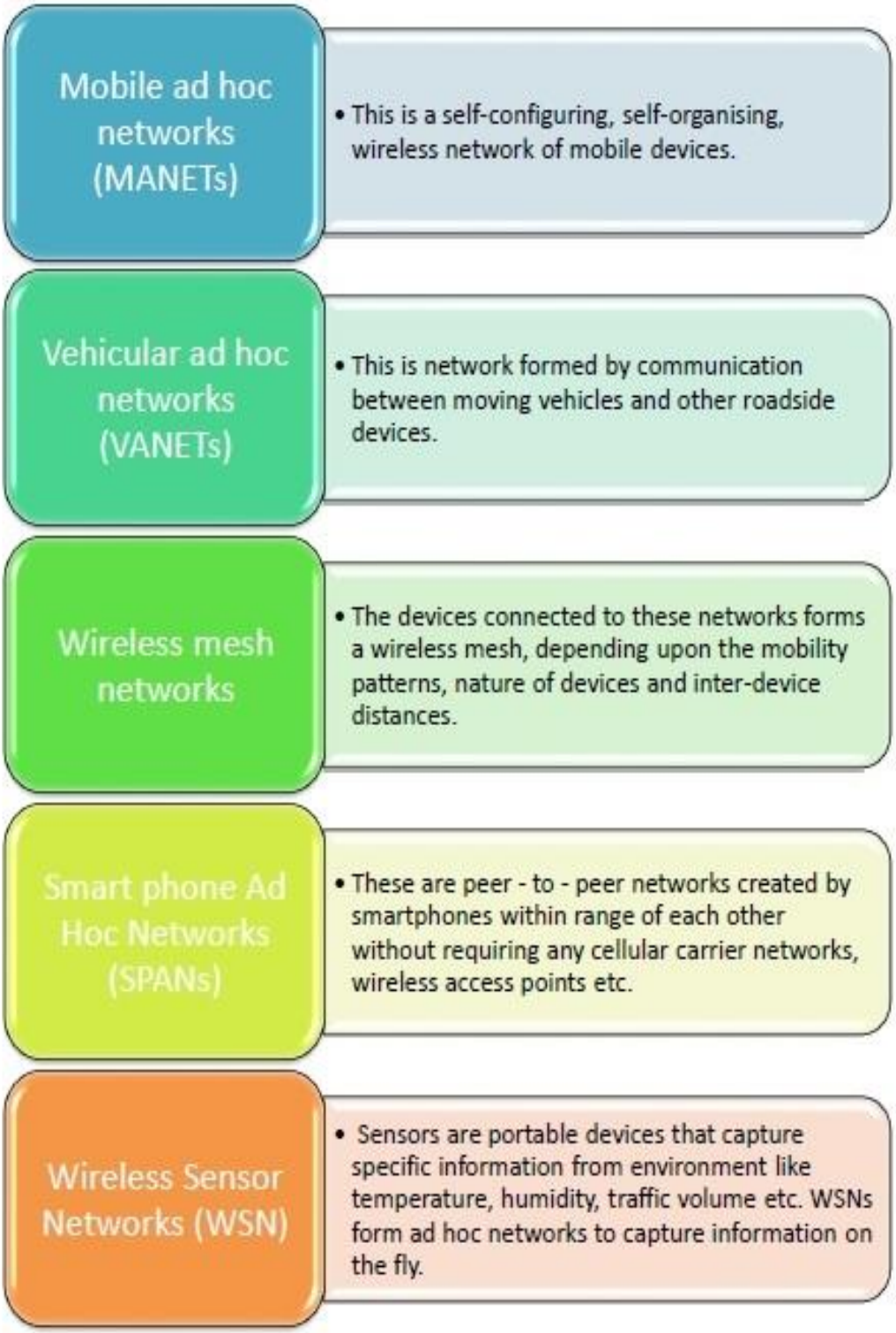
An ad hoc network is one that is spontaneously formed when devices connect and communicate with each other. The term ad hoc is a Latin word that literally means "for this," implying improvised or impromptu.

Ad hoc networks are mostly wireless local area networks (LANs). The devices communicate with each other directly instead of relying on a base station or access points as in wireless LANs for data transfer co-ordination. Each device participates in routing activity, by determining the route using the routing algorithm and forwarding data to other devices via this route.



1.2 Classifications of Ad Hoc Networks

Ad hoc networks can be classified into several types depending upon the nature of their applications. The most prominent ad hoc networks that are commonly incorporated are illustrated in the diagram below −



1.3 Nature of Ad hoc network

In order to understand the potential threats to an ad hoc network, it is important to consider the nature of such networks. The term ad hoc is Latin and it literally means “for this” as in “for this specific purpose.” As the name suggests, ad hoc networks are typically designed for a specific purposes and to work autonomously without having to rely on existing infrastructure. Key characterizations of ad hoc networks are that they can be improvisational, purpose-built, decentralized, and independent. Furthermore, they are typically highly customized for the role for which they are designed to fill. Unsurprisingly, ad hoc networks do not necessarily adhere to a fixed criterion. However, while individual implementations can vary greatly, they can have common features. Such features may include a lack of central control, limited resources, mobility, dynamic topologies, wireless connectivity, and/or custom routing protocols. These features have distinguishable security implications. For example, if an ad hoc network lacks centralized control it will most likely lack a centralized access control mechanism for the network. Wireless networks can be subject to interference and jamming. Additionally, networks with limited resources may not be able to implement typical routing protocols and, therefore, must run a customized protocol. Custom protocols may or may not have sufficient security measures in place depending on the implementation.

1.4 Thread Models

Routing Threats There are a multitude of potential threats facing the routing of ad hoc networks. These include confidentiality, integrity, and availability. When it comes to confidentiality within the realm of routing protocols, the primary threat is towards the, “privacy of the routing data itself.” If the routing data were to be compromised, then a secondary threat could occur to other information such as, “the network topology, geographical location, etc.” The integrity of an ad hoc network essentially relies on the accuracy of each node’s routing information. Potential attacks include those that would either alter existing routing data or introduce new, but incorrect, routing data. Finally, in the context of ad hoc routing, availability fundamentally equates to nodes being able to have on demand access to routing information at all times. Additionally, routing operations should not delay nodes from obtaining up to date information. Consequently, each node within the network should be able to function normally without unnecessary interference from either security or the routing protocol. Additional threat vectors

include authorization, dependability and reliability, and accountability. In its simplest form, authorization refers to whether or not a node is authorized to be on the network. As such, an unauthorized node would be a node that, “is not allowed to have access to routing information, and is not authorized to participate in the ad hoc routing protocol.” Given the nature of ad hoc networks, there may or may not be a formal authorization protocol. Nevertheless, it is still a critical security requirement for access control services. Another common use for ad hoc networks, are wireless communications networks that are used during responses to emergencies. Reliable routing with emergency contingency measures should be implemented in order to ensure dependable and reliable operation. “For example, if a routing table becomes full due to memory constraints, a reactive protocol should still be able to find an emergency route to a given destination.” Finally, accountability of network nodes should also be enforced. If an attack cannot be prevented, then it should be detected. By logging actions that may affect the security of the network, appropriate reactionary measures can be taken. “Event logging will also help provide non-repudiation, preventing a node from repudiating involvement in a security violation.”

External Threats With ad hoc networks, external threats are distinguished from internal threats by classifying external threats as potential attacks performed by unauthorized network nodes or other outside entities. In contrast, internal threats refer to potential attacks originating from internal authorized nodes. In terms of detection difficulty, external threats are typically easier to detect than internal threats. In ad hoc networks with an authentication protocol to block unauthorized nodes from joining the network, external threats typically focus on attacking the data link and physical layers of the network. Also, external attacks can be further classified into two broad categories, passive eavesdropping and active interference. Passive eavesdropping generally refers to attacks that attempt to simply listen to the transmitted signals and network traffic without disrupting the network. The most basic of which simply involves the discovery of a wireless ad hoc networks by detecting the existence of the appropriate signals. By extension, passive eavesdropping can pose a threat to location privacy. More sophisticated attacks will attempt to capture messages, including routing updates. Routing updates can be used to infer the topology of the network and the identities of the more active, and possibly more critical, network nodes. As demonstrated by devices like those in, it is possible to implement passive eavesdropping on wired networks. However, such attacks are typically easier when targeted towards short-range wireless networks that are within close proximity. In contrast to passive eavesdropping, active interference typically involves launching attacks with the aim of service denial. Normally, the denial of service attacks focus on disrupting or distorting communications. Consequently, their effectiveness is governed by the duration of the attack and the routing protocol used by the network. For

example, reactive routing protocols may identify denial of service attacks as line breaks, and trigger an attempt to find an alternative route. On the other hand, proactive routing protocols do not immediately react the non-delivery of packets and instead wait for a connection to time out. Consequently, a denial of service attack may be more effective against networks that employ proactive routing protocols. The most serious type of external denial of service attacks towards ad hoc networks is referred as the, “sleep deprivation torture attack.” These attacks focus on wasting node energy so that it is deliberately wasted. Networks with limited power and resources are typically the most vulnerable. Fortunately, techniques such as spread spectrum technology have been developed to mitigate these types of threats. However, such protections are not effective at the physical layer. Even though limited power is a physical layer constraint, power levels ultimately affect all operations on an ad hoc network. As a consequence, this makes these types of attacks extremely difficult to guard against. However, in addition to power level attacks, there are also threats that target the networks integrity by attempting to change the order of messages or simply replaying old messages across the network in order to expose the network to out of date information. As a consequence, these attacks can effectively delete currently valid routes or trick nodes into using old routes that are no longer valid; effectively disrupting the network.

Internal Threats As mentioned previously, internal threats refer to potential attacks originating from authorized nodes on the network. These types of attacks are potentially very serious since, “internal nodes will have the necessary information to participate in distributed operations.” Typically, the adverse behavior of internal nodes can be classified into four general categories: failed nodes, badly failed nodes, selfish nodes, and malicious nodes. The failed nodes category simply refers to nodes that cannot perform an operation. The badly failed nodes category refers to nodes that behave like failed nodes, but also send out incorrect routing information. The selfish node category refers to nodes that attempt to exploit the routing protocol to their own advantage by not cooperating when a personal cost is involved. Finally, the malicious node category refers to nodes that deliberately attempt to disrupt the network. Furthermore, a node may demonstrate behaviors from multiple categories and multiple nodes within the same category may have differing degrees of incorrect behavior. The potential consequences of adverse node behavior vary from category to category. A failed node that cannot send or forward data packets can affect the security of the network if those packets contain authentication or routing data. Dropped error messages may result in network bottlenecks since the originating node may not be aware that a route is broken and may continue to attempt to use it. The consequence of this behavior can have a more serious impact if the failed node or nodes belong to a secure or emergency route. In comparison to failed nodes, badly

failed nodes pose an additional threat to the integrity of the network. By sending out incorrect routing information, badly failed nodes can unnecessarily increase network bandwidth consumption, cause working links to be marked as broken, and cause neighbor sensing protocols to detect nonexistent neighbors. Furthermore, selfish nodes can also act like failed nodes depending on the operations that they refuse to perform. The main problem associated with selfish nodes is packet dropping. Since many routing protocols are unable to detect when packets are not forwarded, dropped and partially dropped packets can be difficult to detect. As mentioned previously, malicious nodes are nodes that intentionally attempt to disrupt the network. Moreover, the effectiveness of a malicious node is amplified if it is in the position to control the flow of information between other groups of nodes. Malicious nodes can also display any of the other categories of abnormal behavior and, as a result, there are many different potential attacks they can perform. The most common of these attacks is the denial of service attack. The most common technique for denial of service attacks is the sleep deprivation torture attack in which malicious nodes force other nodes to consume their resources by performing unnecessary work. Additionally, denial of service attacks also pose a threat to the integrity of the network by deliberately introducing incorrect routing information. Furthermore, as the density of the node population increases, so does the number of nodes affected. Malicious nodes also pose a threat to networks that utilize neighbor sensing protocols. In this scenario, malicious nodes can force nodes to add nonexistent neighbors or cause other nodes to ignore their neighbors in order to effect a denial of service attack. In addition, malicious nodes can pose as other nodes on the network by using false addresses and disrupt the network's integrity by having other nodes redirect traffic to the malicious node. This technique can be used to perform a black hole attack to capture data or to perform a targeted sleep deprivation attack. In addition to misdirecting traffic, malicious nodes can exploit the route maintenance of the network by propagating false route error messages to mark working links as broken. This can be used to cut a node off from the network or used to force the network to use resources attempting to find alternate routes. Finally, malicious nodes may attempt to use defenses against the network or attack protocol specific optimizations.

1.5 Mechanisms to secure ad hoc networks

Steganography is a technique used to hide sensitive or confidential information within seemingly innocent or unrelated data. It aims to ensure that only authorized users can access the hidden information, while others remain unaware of its existence. Unlike encryption, which focuses on making data unintelligible, steganography focuses on concealing data altogether. In the context of ad hoc networks, where devices communicate directly with each other without relying on a centralized infrastructure, security measures are crucial to protect against various threats. While traditional security approaches primarily focus on encryption and authentication mechanisms at the network or upper layers, steganography provides an alternative method to enhance security. Steganography operates at the link layer of the network protocol stack, allowing it to address specific security requirements independently of the network or upper layers. By embedding information within the link layer frames, steganography can provide strong security services such as confidentiality and authenticity.Confidentiality is achieved by hiding sensitive data within innocent-looking frames. Unauthorized users who intercept the frames will be unable to decipher the hidden information without the knowledge of the steganographic algorithm and the proper decoding process. This ensures that even if an adversary gains access to the transmitted frames, the concealed data remains secure. Authenticity is another vital aspect of secure ad hoc networks. By utilizing steganography at the link layer, it becomes possible to embed authentication data within the frames themselves. This enables devices to validate the authenticity of the received frames and identify potential malicious or spoofed data. One of the advantages of using steganography in the link layer is that it can provide security services independently of the network or upper layers. This means that even if the higher layers of the network protocol stack do not offer robust security measures, the link layer can still provide a level of protection. It reduces the reliance on network or upper layer security mechanisms, which may have vulnerabilities or be susceptible to attacks. However, it's important to note that steganography alone is not a comprehensive security solution. While it can effectively hide information, it does not address other security requirements such as integrity, availability, and non-repudiation. Therefore, a holistic approach that combines steganography with other security measures, such as encryption and authentication at the network and upper layers, is recommended to achieve comprehensive security in ad hoc networks. Additionally, steganography has its own limitations and challenges. Embedding data within the link layer frames can introduce additional overhead and complexity to the network. It may impact the performance and efficiency of the communication, especially in resource-constrained ad hoc network environments. Furthermore, steganography

techniques need to be carefully designed and implemented to withstand various attacks, including statistical analysis and steganalysis, which aim to detect the presence of hidden data. In conclusion, while the link layer of ad hoc networks can offer strong security services through steganography, it should be considered as a complementary approach rather than a standalone solution. By combining steganography with encryption, authentication, and other security mechanisms at the network and upper layers, a more robust and comprehensive security framework can be achieved. It is essential to assess the specific requirements and constraints of the ad hoc network environment and carefully design and implement security measures accordingly to ensure the confidentiality, authenticity, integrity, availability, and non-repudiation of the transmitted data.

2. INTRODUCTION TO STEGNOGRAPHY

Data hiding is of importance in many applications. For hobbyists, secretive data transmission,for privacy of users etc. the basic methods are: Steganography and Cryptography.

Steganography is a simple security method. Generally there are three different methods used for hiding information: steganography, cryptography, watermarking.

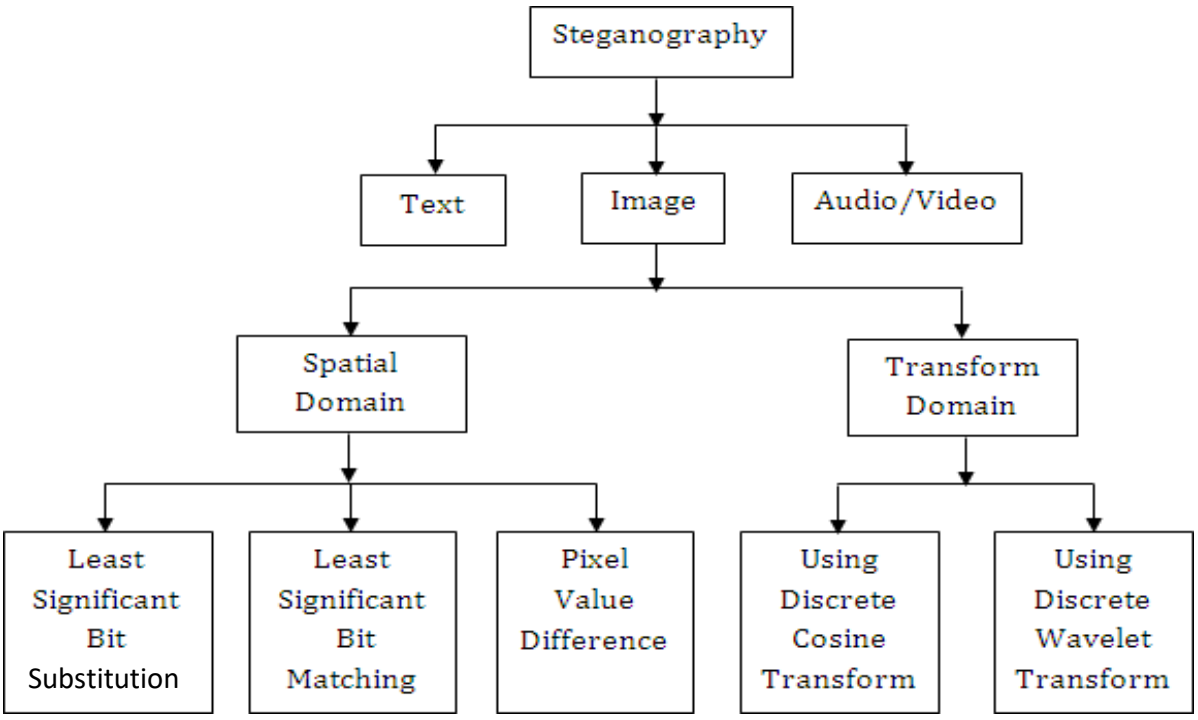
In cryptography, the information to be hidden is encoded using certain techniques; this information is generally understood to be coded as the data appears nonsensical.

Steganography is hiding information; this generally cannot be identified because the coded information doesn't appear to be abnormal i.e. its presence is undetectable by sight.

Steganography is of different types:

1. Text steganography
2. Image steganography
3. Audio steganography
4. Video steganography

In all of these methods, the basic principle of steganography is that a secret message is to be embedded in another cover object which may not be of any significance in such a way that theencrypted data would finally display only the cover data. So it cannot be detected easily to be containing hidden information unless proper decryption is used.



As the above explanation goes, every steganography consists of three components:

1. Cover object
2. Message object
3. Resulting Steganographic object

In this project LSB substitution method is implemented and DCT method is discussed for image steganography. MATLAB is used for coding. The codes and result images are in the following report.

3. TECHNICAL DISCUSSION OF STEGANOGRAPHY METHODS

There are two different methods for image steganography:

1. Spatial methods
2. Transform methods

In spatial method, the most common method used is LSB substitution method.

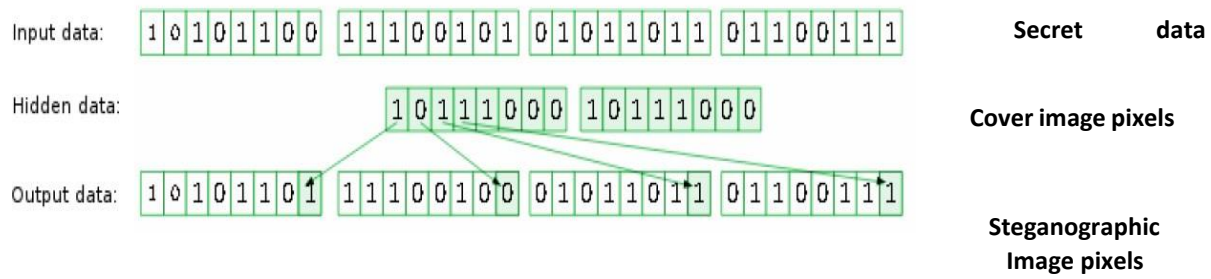
Least significant bit (LSB) method is a common, simple approach to embedding informationin a cover file.

In steganography, LSB substitution method is used. I.e. since every image has three components (RGB). This pixel information is stored in encoded format in one byte. The first bits containing this information for every pixel can be modified to store the hidden text. For this, the preliminary condition is that the text to be stored has to be smaller or of equal size to the image used to hide the text.

LSB based method is a spatial domain method. But this is vulnerable to cropping and noise. Inthis method, the MSB (most significant bits) of the message image to be hidden are stored in the LSB (least significant bits) of the image used as the cover image.

It is known that the pixels in an image are stored in the form of bits. In a grayscale image, the intensity of each pixel is stored in 8 bits (1byte). Similarly for a colour (RGB-red, green, blue)image, each pixel requires 24 bits (8bits for each layer).

The Human visual system (HVS) cannot detect changes in the colour or intensity of a pixel when the LSB bit is modified. This is psycho-visual redundancy since this can be used as an advantage to store information in these bits and yet notice no major difference in the image.



Steps used in LSB steganography:

a. Steps for hiding message image:

1. Read the image to be used as cover image. Noise is added to make it easier

to disguisechanges due to embedding the message image.
2. Read the image to be sued as message image.
3. Separate the bit planes of each image.

As it is known that the LSB (least significant bit) plane contains the least information associatedwith any image, and the MSB (most significant bit) plane contains most of the shape, colour information of an image.

It is generally ideal to replace up to 4 least bitplanes of the cover image, with the upper 4 bitplanes without revealing changes in the resultant image. Lesser number of bitplanes from the message image could be used, but the retrieved image would become distorted and loses information.

4. Replace the least 4 bitplanes of cover image with the 4 most significant bitplanes frommessage image.
5. Get the resultant Steganographic image by recombining these bitplanes.

b. Retrieving message image:

1. Read the Steganographic image.
2. Extract the required number of bitplanes of the image.

Recombining the lower four bitplanes would give the retrieved message image.

4. ALGORITHM OF LBS SUBSTITUTION

1.Convert the image to grayscale

2.Resize the image if needed

3.Convert the message to its binary format

4.Initialize output image same as input image

5.Traverse through each pixel of the image and do the following:

6.Convert the pixel value to binary

7.Get the next bit of the message to be embedded

8.Create a variable temp

9.If the message bit and the LSB of the pixel are same, set temp = 0

10.If the message bit and the LSB of the pixel are different, set temp = 1

11.This setting of temp can be done by taking XOR of message bit and the LSB of the pixel

12.Update the pixel of output image to input image pixel value + temp

13.Keep updating the output image till all the bits in the message are embedded

14.Finally, write the input as well as the output image to local system.

5. CODE (LSB SBUSTITUTION METHOD)

1) %lsb steganography

```
% Read the cover image and set image
coverImage = imread('cover.png');
secretImage = imread('message.jpeg');

% Convert cover image and secret image to grayscale if needed
if size(coverImage, 3) > 1
    coverImage = rgb2gray(coverImage);
end

if size(secretImage, 3) > 1
    secretImage = rgb2gray(secretImage);
end

% Resize secret image to match cover image size
secretImage = imresize(secretImage, size(coverImage));

% Initialize the stego image as the cover image
stegoImage = coverImage;

% Get the dimensions of the cover image
[row, col] = size(coverImage);

% Flatten the secret image into a 1D array
secretArray = secretImage(:);

% Embed the secret image into the cover image using LSB steganography
bitIndex = 1; % Index of the current secret bit to be embedded
for i = 1:row
    for j = 1:col
        % Get the pixel value of the cover image at (i, j)
        coverPixel = coverImage(i, j);

        % Get the bit value of the secret image at the corresponding index
        secretBit = bitget(secretArray(bitIndex), 1);

        % Modify the LSB of the cover pixel with the secret bit
        stegoPixel = bitset(coverPixel, 1, secretBit);

        % Update the stego image pixel at (i, j)
        stegoImage(i, j) = stegoPixel;

        % Move to the next secret bit
        bitIndex = bitIndex + 1;

        % Break the loop if all secret bits have been embedded
        if bitIndex > numel(secretArray)
            break;
        end
    end
```

```
    % Break the loop if all secret bits have been embedded
    if bitIndex > numel(secretArray)
        break;
    end
end

% Save the stego image
imwrite(stegoImage, 'stego_image.png');

% Display the cover image, secret image, and stego image
subplot(1, 3, 1), imshow(coverImage), title('Cover Image');
subplot(1, 3, 2), imshow(secretImage), title('Secret Image');
subplot(1, 3, 3), imshow(stegoImage), title('Stego Image');
```

---

2) **%extracting message - lsb steganography**

```
% Read the stego image
stegoImage = imread('stego_image.png');

% Initialize the secret image as an empty matrix
secretImage = zeros(size(stegoImage));

% Get the dimensions of the stego image
[row, col] = size(stegoImage);

% Extract the LSB of each pixel in the stego image to reconstruct the
secret image
bitIndex = 1; % Index of the current secret bit to be extracted
for i = 1:row
    for j = 1:col
        % Get the pixel value of the stego image at (i, j)
        stegoPixel = stegoImage(i, j);

        % Extract the LSB (secret bit) of the stego pixel
        secretBit = bitget(stegoPixel, 1);

        % Store the extracted secret bit in the secret image at the
corresponding index
        secretImage(bitIndex) = secretBit;
```

```
        % Move to the next secret bit
        bitIndex = bitIndex + 1;

        % Break the loop if all secret bits have been extracted
        if bitIndex > numel(secretImage)
            break;
        end
    end

    % Break the loop if all secret bits have been extracted
    if bitIndex > numel(secretImage)
        break;
    end
end

% Reshape the secret image to match the size of the original secret image
secretImage = reshape(secretImage, size(stegoImage));
% Convert the secret image to grayscale if needed
if size(stegoImage, 3) > 1
    secretImage = mat2gray(secretImage);
End
% Display the secret imageimshow(secretImage);
title('Decoded Secret Image');
```

6. TRANSFORM DOMAIN TECHNIQUES ALGORITHM

6.1 Embedding Algorithm:

1. Read the cover media (e.g., image) and obtain its transformed representation (e.g., apply Discrete Cosine Transform (DCT) or Discrete Wavelet Transform (DWT)).
2. Convert the secret data into a binary representation.
3. Determine the coefficients or features in the transformed domain to be used for embedding the secret data.
4. Iterate through the transformed coefficients/features and the binary secret data.
   a. Modify the selected coefficient/feature based on the secret data.
   b. Adjust the modified coefficient/feature to maintain the statistical properties of the cover media (optional).
5. Reconstruct the transformed representation using the modified coefficients/features.
6. Save the stego media (e.g., stego image).

6.2 Extraction Algorithm:

1. Read the stego media (e.g., stego image) and obtain its transformed representation.
2. Determine the coefficients or features in the transformed domain from which to extract the secret data.
3. Iterate through the selected coefficients/features.
   a. Extract the bit(s) from the coefficient/feature.
   b. Accumulate the extracted bits to form the binary representation of the secret data.
4. Convert the binary secret data to its original form (e.g., text or image).

7. CODE (TRANSFORM DOMAIN TECNIQUE)

1) Embeds secret_data into image using DCT-based Transform Domain Techniques

```
function transform_domain_embed(input_image, output_image, secret_data,
alpha)
    % Read the input image
    image = imread(input_image);

    % Convert secret_data to binary
    secret_data_binary = dec2bin(secret_data, 8);
    secret_data_binary = secret_data_binary(:) - '0';  % Convert to column
vector of bits

    % Perform DCT on image blocks
    dct_image = blockproc(image, [8 8], @(block_struct)
dct2(block_struct.data));

    % Embed the secret data in the DCT coefficients
    num_coeffs = numel(dct_image);
    if length(secret_data_binary) > num_coeffs
        error('Insufficient capacity to embed the secret data.');
    end

    coeffs = reshape(dct_image, 1, []);
    coeffs(1:length(secret_data_binary)) = ...
        coeffs(1:length(secret_data_binary)) + alpha * secret_data_binary;

    % Reshape the modified coefficients back to the DCT image size
    modified_dct_image = reshape(coeffs, size(dct_image));

    % Perform inverse DCT to get the modified image
```

```
    modified_image = blockproc(modified_dct_image, [8 8], @(block_struct)
idct2(block_struct.data));

    % Convert the modified image back to uint8
    modified_image = uint8(modified_image);

    % Write the modified image to the output file
    imwrite(modified_image, output_image);
end
```

% Extracts secret_data from image using DCT-based Transform Domain Techniques

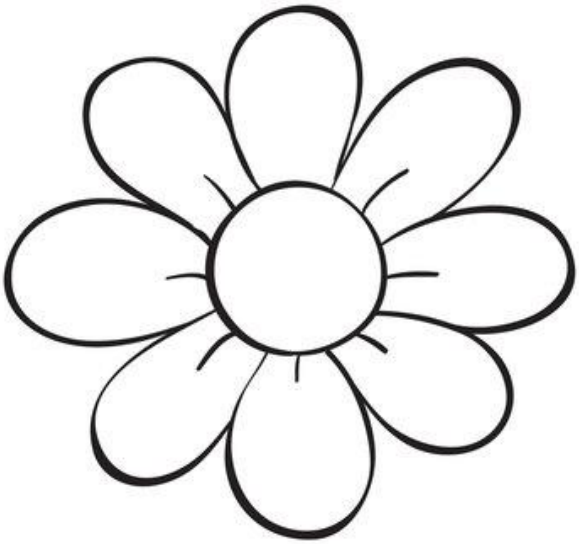
```
function extracted_data = transform_domain_extract(input_image, num_bits)
    % Read the input image
    image = imread(input_image);

    % Perform DCT on image blocks
    dct_image = blockproc(image, [8 8], @(block_struct)
dct2(block_struct.data));

    % Extract the secret data from the DCT coefficients
    coeffs = reshape(dct_image, 1, []);
    extracted_data = bitget(coeffs(1:num_bits), 1);
    extracted_data = char(bi2de(reshape(extracted_data, 8, [])'))';
end

% Example usage
input_image_file = 'input_image.jpg';
output_image_file = 'output_image.jpg';
secret_message = 'This is a secret message.';
alpha_value = 0.1;

% Embed the secret message in the image
```

```
transform_domain_embed(input_image_file, output_image_file, secret_message,
alpha_value);


% Extract the secret message from the modified image

extracted_message = transform_domain_extract(output_image_file,
numel(secret_message) * 8);

disp('Extracted message:');

disp(extracted_message);
```

8. RESULT

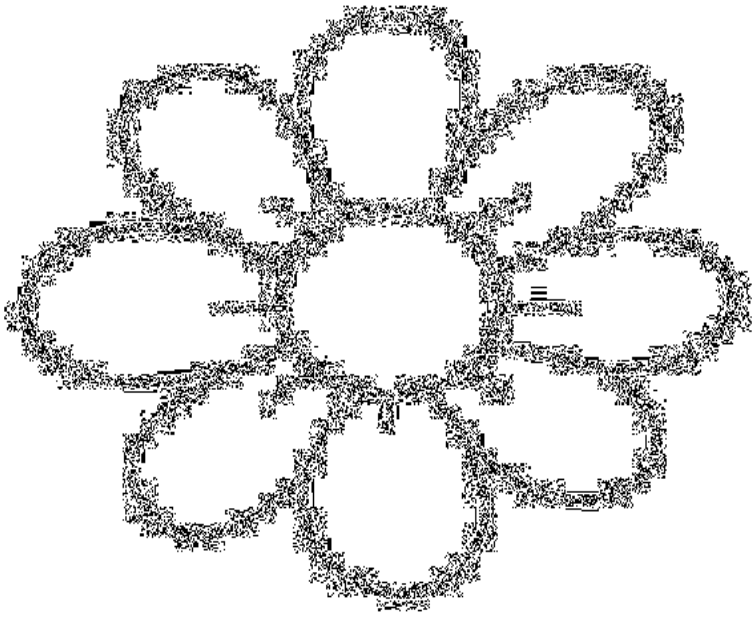
Cover image

Message image



steganographic image

Extracted message image



CONCLUSION

It is observed that to provide security to ad hoc network through LSB Substitution Steganographic method and DCT method is pretty great, no need to go through OSI model to improve security, in steganography the results obtained in data hiding are pretty impressive as it utilizes the simple fact that any image could be broken up to individual bit-planes each consisting of different levels of information. It is to be noted that as discussed earlier, this method is only effective for bitmap images as these involve lossless compression techniques. Also, in this project grey-scale images have been used for demonstration. But this process can also be extended to be used for color images where, bit- plane slicing is to be done individually for the top four bit-planes for each of R, G, B of the message image, which are again to be placed in the R, G, B planes of the cover image, and extraction is done similarly.

It can be observed that the result image after extraction is not very clear as the initial message/secret image. This can be explained by the fact that this is a very high resolution image. From, it is seen that data is visible in the message image in planes 5, 6, 7 to the human eye, and rest of the bit-planes appear to be dark/ black, but these also have tiny bits of information which is ignored for the process of data hiding. Better results can be obtained if the message image to be hidden is of a lower resolution. This is observed from Image set 2, where the extracted image has less loss compared to the original messageimage.

It is also important to discuss that though steganography was once undetected, with the variousmethods currently used, it is not only easy to detect the presence but also retrieving them is easier. For instance, without having to use a software or complex tools for detection, simple methods to observe if an image file has been manipulated are:

1. Size of the image: A Steganographic image has a huge storage size when compared toa regular image of the same dimensions. I.e. if the original image storage size would befew KBs, the Steganographic image could be several MBs in size. This again varies with the resolution and type of image used.
2. Noise in image: A Steganographic image has noise when compared to a regular image.This is the reason why initially little noise is added to the cover image, so that the Steganographic image doesn't appear very noisy when compared to the original coverimage.

Though this project focusses on LSB and spatial domain steganography, few details about transform domain methods have also been researched, basics of

which have been discussed. So through the various articles and theory available, it is observed that transform domain methods perform better in comparison with spatial domain methods.

REFERENCES

[1] N. Meghanathan, "Stability and Hop Count of Node-Disjoint and Link-Disjoint Multi-path Routes in Ad Hoc Networks," Proceedings of the 3 rd IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, New York, October 2007.

[2] B. Liang and Z. Haas, "Predictive Distance-based Mobility Management for PCS Networks," Proceedings of the IEEE International Conference on Computer Communications, Vol. 3, pp. 1377- 1384, March 1999.

[3] T. Morkel 1, J.H.P. Eloff 2, and M.S. Olivier 3, an overview of image steganography, Information and Computer Security Architecture (ICSA) Research Group.

[4] Walaa Abu-Marie, Adnan Gutub and Hussein Abu-Mansour, Image based steganography using Truth Table Based and Determinate Array on RGB Indicator, International Journal of Signal and Image Processing (Vol.1-2010/Iss.3) Abu-Marie et al. / Image Based Steganography Using Truth Table Based and Determinate … / pp. 196-204

[5] Soumyendu Das, Subhendu Das, Bijoy Bandyopadhyay And Sugata Sanyal, Steganography and Steganalysis: Different Approaches, Available from: http://arxiv.org/ftp/arxiv/papers/1111/1111.3758.pdf

[6] Aumreesh Kumar Saxena, Sitesh Sinha, Piyush Shukla. Design and Development of Image Security Technique by Using Cryptography and Steganography: A Combine Approach. International Journal of Image, Graphics and Signal Processing. 2018; 10(4): 13-21.