Abstract
Steganography techniques are associated with a unicast relation, in which one sender uses a digital medium to deceptively hide a secret message intended for a single receiver. However, the frequency of sending cover images can lower the imperceptibility of the communication and arouse the suspicion of a lurking eavesdropper. In this paper, we extend the notion of unicast steganography to one in which the sender is capable of sending a single digital cover medium containing multiple messages to multiple receivers and each message is securely intended for a sub-group of the receivers. We present a shared medium scheme, Shared Medium Steganography (SMS), which minimizes the exchange of cover images while maintaining the security and privacy of the individual messages.

Keywords: Covert Communication, Encryption, Multiple Receivers, Shared Message, Steganography

INTRODUCTION
Since audio and video first became available in digital form, people have been motivated to communicate secretly by embedding secret messages in these mediums (Katzenbeisser 2000). Such acts are known as steganography, which is defined as the “art and the science of invisible communication” (Morkel 2005). There are many different digital content types that can be used as a cover for a secret message. Images, videos, and audio are three popular types used in steganography due to their high level of redundancy. Digital redundancy is the superfluous bits that provide much more visual accuracy to be noticed by the human visual system (HVS) such that slight changes to these bits do not result in any noticeable distortion (Batra 2013). These redundant bits provide a perfect environment to hide a secret message.

Steganography can be considered a tool for establishing covert communication between two parties. However, it is currently applied to one-to-one communication, where one sender deceptively hides a message in a digital medium and sends it to a single receiver. However, it is insufficient to only hide the secret message; not arousing any adversary suspicion is also necessary. An adversary can detect a steganographic secret communication by checking the quality of cover images being sent, with images of lower quality hinting at changed bits and possibly a hidden message. The number and frequency of cover images sent can also fuel suspicion. Therefore, it is imperative that any effective steganography-based communication scheme endeavor to reduce the effect that it has on cover image quality and the number of cover images sent to the receivers.

Since steganography was first presented and used in digital mediums, we have witnessed diverse approaches and techniques in literature to establish a unicast steganographic communication of the form of one sender/one receiver/one message but not in the form of a shared medium wherein several messages are sent on a single cover image. A shared message steganographic communication is of interest because it has the benefit of reducing the number of cover images being sent to different receivers and thereby lowering the suspicion that an adversary might have. The only caveat is that all recipients must know the embedding techniques and processes used to recover the secret messages. An example would be a bank-generated cover image that embeds the information of multiple credit cards of a bank customer, in which a point of sale reader would scan the cover image to recover only the information of the customer’s selected credit card. In shared-medium steganography, it is essential that messages be read by only the intended readers and that the secrecy of the other messages on the cover image is guaranteed.

In this paper, we present and analyze a novel and efficient scheme to covertly send a single cover medium to a group of receivers that contains multiple secret messages, each of which is intended for a sub-group of the receivers. We believe that there is no work in the literature that articulates this shared medium steganography problem and provides a comprehensive solution for it.

The rest of the paper is organized as follows. Related work is summarized in section 2, and the shared medium steganography scheme is presented in section 3. In Section 4, the experimental results of the proposed scheme are presented and discussed. Finally, we conclude the paper with conclusions and future work.

RELATED WORK
Steganography embedding techniques can be broadly classified into two domains: the spatial domain and the frequency domain. In each domain, several embedding techniques have been proposed in the literature (Johnson 2001, Wu 2013, Bailey 2006, Muhammad 2015, Parvez 2008, Gutub 2010, Bandyopadhyay 2014).

Research has been performed on increasing the security of the embedding process beyond that of just steganographic insertions. In (Cheddad 2008), a novel scheme for the protection of two-dimensional data, such as images, is presented. The scheme uses an extended version of SHA-1 as a pre-process phase before using the image as a cover image.
to embed secret messages. Several works have added a layer of security to steganography by encrypting the message in a pre-process phase before embedding the message into the cover image. Juneja et al. (Juneja 2009) proposed an LSB insertion-based technique that encrypts the secret message using RSA key encryption before embedding the secret message in the cover image. Ramaiya et al. (Ramaiya 2013) proposed to preprocess the secret message with DES encryption and S-Box mapping before embedding it in the cover image. Sanini et al. (Sanini 2013) introduced a scheme where the secret message is pre-processed by a new, modified version of AES encryption before embedding. Charan et al. (Charan 2015) presented a scheme that applies multi-layer encryption to the secret message before embedding in the cover image. Their scheme pre-processes the secret message in two stages: Caesar Cipher is used in stage one and the message is encrypted using Chaos Theory in stage two. Steganography grew in other directions in addition to embedding secret text into a cover image. Yadav et al. (Yadav 2013) proposed LSB insertion-based video steganography, where a secret video stream is embedded in another cover video stream. In (Ogiela 2015), Ogiela et al. proposed to embed both a secret message and a false message in the cover image. The false message would serve as a decoy for an eavesdropper. They proposed to use a secret key to create a permutation of pixels where the secret message is to be embedded, scattering it all over the cover image, while the false message is embedded in the cover image using the LSB insertion technique. Liu et al. (Liu 2015) introduced a scheme to send a secret message to multiple receivers using quantum steganography. They built a hidden communication channel in HBB through the entanglement swapping of Greenberger-Horne-Zeilinger (GHZ) states and Bell measurement. Their scheme is dependent on the honest participation of all of the intended receivers to recover the secret message and is immune to an external eavesdropper.

**SHARED MEDIUM STEGANOGRAPHY (SMS)**

The problem that we are attempting to solve is how to embed several secret messages on a single cover digital medium and send the cover image to several recipients, with each message intended for one or more of those recipients, as shown in Figure 1, while protecting the secret messages from being read by unintended recipients.

Let x be the number of secret messages to be sent and y be the number of intended receivers of all x messages. We propose that a cover image be divided into x partitions, with one partition allocated exclusively for one message to be embedded. Let us define the recipient to be the intended receiver of the cover image containing all xsecret messages. Let us also define the intended reader of message i to be the receiver who is permitted to view the embedded secret message i in the cover image. Hence, the relationship among different variables in our scheme can be summarized as follows:

∀ CoverImage ∃ y recipients
∀ message_i ∃ 1 ≤ number of Readers ≤ y

We define a set \( R_i \) to be the set of all intended readers of the message.

\[ R_i = \{ \text{recipient}_j \} \]

where recipient_j is a reader of message i. We also define a set \( R \) to be the set of all recipients of a cover image.

\[ R = \bigcup_{i=1}^{x} R_i \]

\[ |R| = y \text{ and } 1 \leq |R_i| \leq y \]

It is imperative that we protect any secret message embedded in a cover image from the eyes of those recipients who are not intended readers. To accomplish this objective, our proposed scheme uses the following steps.

**Figure 1:** The three traditional communication types and shared medium

**Step 1: Message Location**

Because the cover digital medium to be used may contain multiple secret messages intended for several receivers, it is imperative to establish a mechanism to allocate distinct parts of the cover medium for each distinct secret message to be embedded. Essentially, each message is allocated a smaller cover image for its own embedding. For example, when using colored-image LSB insertion, each message will have a unique combination of pixel number and channel bits. We can virtually separate the messages by first partitioning the cover digital medium, and each single message is exclusively allocated a partition, as shown in Figure 2. The number of partitions into which the cover medium will be divided and the size of each partition are assumed to be agreed upon among all receivers before the communication commences, not during the communication.
Step 2: Message Privacy

In shared medium communication, each message is intended to be read by a group of selected receivers. Hence, we need a mechanism to secure the message from unintended receivers who also happen to be readers of other messages embedded in the cover medium. For example, in the shared medium communication illustrated in Figure 1, the sender sends message 1 to receivers 1 and 2, message 2 to receiver 3, and message 3 to receivers 4, 5, and 6. These messages are embedded in one stego cover medium. Therefore, message security must be applied to prevent unintended receivers from uncovering the other secret messages on the same cover image (e.g., receiver 1 decoding message 3).

To establish secure embedding, we propose to encrypt the secret message before embedding, which will brand the system as a hybrid of steganography and cryptography, using one of the popular and accepted encryptions, such as AES, RSA, or Blowfish. This will guarantee that only the intended readers are able to recover the secret message. However, this poses an interesting challenge because the sub-groups of intended readers may be different for all communicated messages from one communicated cover image to the next. For example, take the case where in the first cover image, receivers 1 and 5 share a message and receivers 2, 3, and 4 share another message. In the next cover image sent using our proposed scheme, receivers 3 and 4 might share a message, receivers 1 and 5 might share another message, and receiver 2 might be the sole intended reader of a message. This dynamically changing sub-grouping of intended readers presents a challenge in how encryption keys are distributed and used.

To overcome this challenge, we propose to have a key for each combination of receivers. Table 1 shows an example for 3 users from the perspective of user 1. In each row, 1 and 0 represent a key that is shared between that user or is not shared, respectively. For example, Key 3 is used to encrypt a message shared among all three users (indicated by 1 for all receivers). Because we are following user 1’s perspective, all of the keys in the first column are 1.

Table 1: Keys stored at user 1

<table>
<thead>
<tr>
<th>User 1</th>
<th>User 2</th>
<th>User 3</th>
<th>Key Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>Key 1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Key 2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Key 3</td>
</tr>
</tbody>
</table>

If we look at the readers as combinations, then we can see that the number of keys each receiver must possess follows the following formula:

Number of shared keys stored in each receiver

\[
\sum_{r=1}^{n} \binom{n}{r} - \sum_{r=1}^{n-1} \binom{n-1}{r} - 1
\]

(1)

where \( n \) is the total number of receivers and \( r \) is the number of intended readers in SMS. Because we are calculating the number of keys receiver \( i \) must possess, in Equation 1, we subtract the combinations that do not involve receiver \( i \) from the total. Alternatively, we can simply look at it as the number of subsets that involves the intended user from a set of all users (e.g., the number of subsets containing user 1 from subset \{user1, user 2, user 3\}), which is equal to the number of subsets that can be made from all other users, as shown in the following formula

Number of public keys stored in each user end = \((2^n-1) - 1\)

(2)

Therefore, a receiver that is not intended to view the message will not be able to do so because the message is encrypted by a key that the unintended receiver does not possess. For example, if we want to send message 1 to receivers 1 and 3 only, we use Key 2 (see table 1). This key is only possessed by receivers 1 and 3; therefore, the other receivers are not able to decrypt the message.

With the exponential increase in the number of keys to be stored at the receivers as the number of receivers increases, it is more efficient to consider the selected key distribution instead of considering all possible combinations of receivers. In practical situations, message sharing is governed by a set of rules that are set forth by an organization, and the distribution of keys would be performed only for the set of receivers who need to communicate secret messages. Such key distribution is called selected distribution and would reduce the number of keys stored at the receivers.

The assumption here is that the appropriate keys are created and distributed to all receivers before the commencement of communication; therefore, adding new receivers later would require a new communication with all receivers as to the additional keys.
Because as many as \(2^{n-1} - 1\) keys may be stored at each receiver, it is impractical for the receivers to use each key to try to decode the secret message. Therefore, a more efficient method of communicating which key is used to encrypt a secret message is needed for the sender to inform the intended readers of which key they need to use to recover the secret message. A trivial solution would be to communicate the key information directly to intended readers through a separate channel over the network. There is an argument against such a solution because any extra communication outside the covered medium would give eavesdroppers an opportunity to recover the key information [17] or at least indicate a relationship between the communicating parties, which is intended to be kept secret.

We propose to allocate \(Z\) bits, where \(Z\) is given by Equation 3, at the start of every cover image partition to embed the key id of the encryption key used to encrypt the secret message embedded in this image partition.

\[
Z = \lceil \ln(n) \rceil
\]  

Upon receiving the cover medium, all receivers read the first \(Z\) bits of every partition to determine which encryption key to use to decrypt the secret message. Intended readers will find a key id that they already possess, whereas other receivers will not. Using this mechanism, receivers can determine how many messages they are intended to read and the location of these messages by checking the key ids embedded in the \(Z\) bits of every partition and comparing them with the keys they possess. This is an efficient, embedded solution to notify the intended readers of what and where to read the secret messages without requiring a separate communication tool to relay this message related information.

**Step 3: Embedding Technique**

There are two basic classes of embedding techniques in steganography: spatial embedding and frequency embedding. Although our proposed scheme can be applied with either class’s techniques, our proposed scheme uses the popular spatial domain technique least significant bit (LSB) to embed the secret message. Figure 3 illustrates and summarizes the main steps performed to establish the cryptographic LSB-based shared medium steganography communication. The figure illustrates the case of sending three secret messages to 7 intended readers of 10 total receivers in the system. Message 1 is intended for receivers 1 and 2, message 2 is intended for receiver 3, and message 3 is intended for receivers 4-7. The first step of the scheme involves partitioning the cover medium (in this case an image) into \(p=16\) partitions. The three secret messages are then secured by encrypting them with the appropriate shared keys. For example, for message 1, the key used is the one shared by receivers 1 and 2 only. Each encrypted messages is then embedded into a selected partition using traditional LSB embedding. Because the payload is already encrypted, there is no need to use secure LSB embedding techniques. After embedding all of the secret messages in the cover image, the key ids used to encrypt each message are embedded in the \(Z\) bits of the partition they were embedded in. Finally, the stego image is produced and sent to all of the receivers.

**SECURITY ANALYSIS**

In basic steganography work, the body of research considers only lurking adversaries trying to uncover the secret messages embedded within the cover images. These adversaries are not among the receivers because steganography work has concentrated on unicast, i.e., one sender one receiver, communication. However, in our proposed scheme, where a cover image is shared among several receivers, groups of whom are the intended readers of the secret messages, whereas others are not, there is the possibility of one or more of the unintended receivers becoming curious adversaries trying to uncover one or more messages of which they are not the intended readers. There are two curious adversaries from whom our proposed scheme needs to protect the secret messages: outsider adversaries and unintended receivers.

Our proposed scheme poses a greater challenge for an outsider adversary than is posed by basic steganography schemes in the literature because an outsider adversary does not know how the cover image is partitioned, how many partitions there are, the partitions in which the messages are embedded, or the key that is used to encrypt the embedded messages. It would be infeasible for an outsider-adversary to brute-force these factors to uncover the embedded secret messages. A receiver who is an unintended reader poses a different threat to the concealed message. Unlike the outsider-adversary, a receiver in our proposed scheme has knowledge of the number of partitions in the cover image, the partition sizes, and what embedding technique is used. Hence, a receiver knows where the messages are. However, each secret message is encrypted by a specific encryption key whose id is embedded in the \(Z\) bits of the partition in which the secret message is embedded. Embedding the encryption key ids in the \(Z\) bits as described will make them accessible to read by all receivers, not just the intended readers. Even if the receivers do not possess the key referenced by the embedded key id, the receiver might still infer that there is a secret message that they are not intended to read, which would arouse their suspicion and curiosity regarding what key was used and for whom this message was intended. We present a counter argument that even if an unintended receiver reads the \(Z\) bits and determines the encryption key id, this unintended receiver would not possess the appropriate key and would thus have to guess what key is being used. Because we are using standard shared key encryption algorithms, the curious unintended receiver will have to brute-force trial all possible...
keys, which is similar to any adversary trying to decrypt an encrypted message on the Internet. The robustness of message encryption can be increased by increasing the key size, which will make it infeasible for an adversary to brute-force the key. The curious unintended reader will not know who the intended readers are because only relevant keys are stored. Moreover, keys are not generated in any pattern that a curious unintended reader might use to arrive at the intended readers of a message. Because we are using the LSB technique to embed the payload, we compare the traditional LSB embedding and our proposed method. Table 2 summarizes the features of each system.

Table 2: Comparison between traditional LSB and the proposed scheme

<table>
<thead>
<tr>
<th>Traditional LSB</th>
<th>Proposed Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication Type</td>
<td>Cover medium</td>
</tr>
<tr>
<td>Number of Receivers</td>
<td>Single</td>
</tr>
<tr>
<td>Number of Messages per Cover Medium</td>
<td>Single</td>
</tr>
<tr>
<td>Intended Readers per Message</td>
<td>Single</td>
</tr>
<tr>
<td>Embedding Overhead</td>
<td>No</td>
</tr>
<tr>
<td>Communication Overhead</td>
<td>No</td>
</tr>
<tr>
<td>Complexity</td>
<td>Simple Embedding</td>
</tr>
</tbody>
</table>

From Table 2, we can see that our propose scheme provides many features compared with a basic LSB insertion scheme, though with some added complexity and embedding overhead. The complexity required is not extravagant and is well within the basic computing power of today’s personal computing machines. The embedding overhead caused by embedding the message encryption keys in the Z bits of each partition is also tolerable, considering that the scheme only embeds a key id that is only a few bits. For example, with the cover medium partitioned into 16 partitions and with 32 receivers, the total embedding overhead would be 80 bits.

EXPERIMENTAL RESULTS
To investigate the effectiveness, we implemented our proposed scheme, setting the number of receivers in our SMS system to five and the number of messages to be communicated to three. The RSA algorithm was used to encrypt the three messages to establish the secure embedding. With 5 users in this SMS system, the total number of public keys is 31, as given by Equation 1. However, the number of public keys stored at each receiver is only 15, as given by Equation 2. As a test case, we assumed that user 1 is set to send three messages to recipients, as shown in Table 3. Messages 1 and 3 are unicast communications for recipients 2 and 4, respectively, whereas message 2 is a shared message between recipients 3 and 5.

Table 3: Receiver (users 2, 3, 4 and 5) details for our three messages sent from user 1

<table>
<thead>
<tr>
<th>Message Number</th>
<th>Recipient Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>2</td>
<td>0 1 0 1</td>
</tr>
<tr>
<td>3</td>
<td>0 0 1 0</td>
</tr>
</tbody>
</table>

The image quality metric peak signal-to-noise ratio (PSNR) is measured for the implemented scenario. Each of the three messages is 100 characters long. The three messages are encrypted using the RSA algorithm and the appropriate public key. The SMS scheme is compared with traditional LSB embedding of the concatenation of the three messages in a single cover medium to investigate the possibility of overhead from implementing the proposed SMS scheme. Table 4 presents the PSNR results obtained using five different cover images shown Figure 4.

Table 4: Proposed scheme vs. traditional method

<table>
<thead>
<tr>
<th>Cover Image</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cryptographic LSB</td>
<td>Traditional LSB</td>
</tr>
<tr>
<td>Mandrill</td>
<td>75.384798</td>
</tr>
<tr>
<td>Peppers</td>
<td>75.470741</td>
</tr>
<tr>
<td>Airplane</td>
<td>75.220677</td>
</tr>
<tr>
<td>Colored Birds</td>
<td>77.064315</td>
</tr>
<tr>
<td>Sailboat on Lake</td>
<td>75.332297</td>
</tr>
</tbody>
</table>

In both cases, the number of message bits to be embedded is exactly the same. As such, the recorded PSNR values are relatively similar, and the slight difference in the PSNR results shown in Table 5 is due to the difference in the embedding area. Recall that in our proposed scheme, we embed each message in a separate partition in the cover medium, whereas a traditional LSB implements from the first pixel until it finishes. The results in Table 4 indicate that we can achieve the same quality of cover medium as a traditional LSB embedding with the added advantage of sending multiple messages to multiple sub-groups of multiple receivers in one cover medium.
CONCLUSION

In this paper, we have presented a scheme called shared medium steganography that extends the rigid unicast steganography communication notion by going beyond the traditional one-to-one information hiding communication to a one-to-many type of steganographic communication. Moreover, the proposed scheme allows for multiple messages to be intended for different sub-groups of receivers while being embedded in the same cover medium.

To achieve such a shared medium communication in steganography, certain attributes are required to secure each embedded message from unintended readers. Moreover, key management and distribution is another important factor to consider. Our proposed scheme addresses these requirements and provides a secure means of communication by partitioning the image and embedding each secret message into a single unique partition. The proposed scheme uses encryption to establish message security before embedding while distributing the keys in a manner that preserves the privacy of the message. The proposed scheme was implemented and tested, with recorded PSNR results that are high, pass the HVS and are similar to those of traditional LSB embedding.

COMPLIANCE WITH ETHICAL STANDARDS

The authors assert that there is no financial or non-financial conflict of interests with regards to this work and manuscript that is not under consideration or submitted to any other journal. The authors also affirm that no human or animal subjects were used while conducting this work.

REFERENCES


