An Experimental Study to Evaluate the Integration of Various Security Approaches to Secure Transferable Data

Mohammad Shkoukani [1], Ahmad Mousa Altamimi [2], Hazem Qattous [3]

[1] *Department of Computer Science*, m.shkokani@asu.edu.jo
[2] *Department of Computer Science*, a_altamimi@asu.edu.jo
[3] *Department of Software Engineering*, hqattous@asu.edu.jo

Applied Science Private University, Amman, Jordan.

***Abstract* - This study investigates the relative ability of combining various security techniques to increase the security level of transmitted data without affecting the cover file. The analysis relies on an empirical study that evaluates the capability of integrating three different but related security measures: encryption, stenography and compression, in a combined manner. These proved security measures are customized in order to provide adequate security to the data before outsourcing it. To validate our proposed technique, a security framework is implemented and evaluated using several different text and audio wave files of 16 bits per sample. The results show that hiding data using 4 bits does not show any difference between the original and stego file. In addition, the One-Time-Pad algorithm proved that it allows a larger space to be available to hide data rather than using Pohlig-Hellman algorithm.**

Keywords - security; cryptography; steganography; compression; combining security

I. INTRODUCTION

Securing information is one of the most challenging issues in today's technological world. To secure the transmission of secret data over the Internet, various schemes have been presented over the last decade [1-2]. One can consider in this regard the two common countermeasures: cryptography and steganography. Cryptography is the science of securing messages to provide various aspects such as data confidentiality, integrity, authentication, and non-repudiation [3]. It is old as Julius Caesar who invented a way (called Caesar cipher) for encrypting his political and military communication in 50 BC. In fact, the history also tells about the importance of using cryptography. For example, during the World War II, both sides were worked hard to introduce new encryption techniques that the other side could not discover or decrypt [4].

On the other hand, Steganography, historically, comes from two Greek words, steganos, which means the covered or secret and graphy, which means writing or drawing and defines as the process of hiding one communication medium into another [5]. Steganography usages extend to include different applications including watermarking and digital rights and annotation in addition to other applications. One of the most important applications for steganography is covert communications [6].

The main difference between steganography and cryptography is that in steganography, the data is modified. While, in cryptography, the data is modified and transformed into another form or format to prevent its understanding unless it is decrypted using specific manner and keys [7]. It is important to mention here that either technique alone could secure the data. However, combining them together can increase the security of the message. In fact, combining of cryptography and steganography is suggested by many authors as [8-10].

Cryptography and steganography could be combined by encrypting the message then hiding it. Although the idea has been discussed and suggested previously, its application is limited until now because such an idea needs an extensive research and studies to investigate different encryption techniques with different steganography techniques. For example, the work of [11] introduced the idea of combining cryptography and steganography with providing an implementation.

On the other hand, data compression has also considered in securing the data [12-13]. It defines as the production of a compressed version of an input data, which is smaller in size and holds the same information that the original has. Based on that, authors employed the compression process to convert the format of data to hide data. Recently, Authors of [13] proposed an engine for securing and managing of 3D medical images by compressing them before transmission.

Being said that, our study is differentiated from the above works in many aspects. Firstly, a research has been conducted to find the desired choice of the steganography technique to be used. Secondly, the visibility of combining cryptography and steganography is explored to produce a higher level of security. Thirdly, the effect of applying our technique on the host file that will be used as a carrier file is measured to determine if there is any difference between the original and the stego file (which contains hidden data) for

the entire tested sample. Finally, the problem of keys exchange is also investigated by means of the ability to transfer encryption secret keys safely without the need for using secure channel.

To underscore the practical viability of our approach, an empirical study was conducted to highlight the differences between the original file and the stego file, where the system is evaluated using text file and several different wave files of 16 bits per sample of different sound styles. 1 bit, 2 bits, 3 bits, 4 bits, 5 bits and 8 bits are used to hide data into each of the tested sample. The results demonstrate that hiding data using 4 bits does not show any difference between the original and stego file.

The rest of the paper is organized as follows. Section II presented the reviewing of some related works to the proposed approach. Background material that is related to our research is introduced in Section III. While the system development and evaluation are presented in Section IV. Section V discuss the results, while Section VI offers the final conclusions and future works.

II. RELATED WORK

Many techniques are found in the literature appropriate for securing the transmitted data. In fact, the encryption techniques have been widely used to develop crypto systems [13-14]. On this subject, the two main techniques of cryptography- Symmetric and Asymmetric were utilized. While the symmetric cryptography (e.g., DES, AES, Pohlig-Hellman, One-time pad, and others) uses the same key used for encryption and decryption, the Asymmetric Key encryption (e.g., RSA, ElGamal, Diffie–Hellman, and others) uses two different keys, Public key for encryption and private key for decryption [14]. In fact, many research works have been developed as literature surveys to focus on the different kinds of encryption techniques that are existing [14-16].

Steganography, on the other hand, has also considered to hide information as a complementary security solution. There are several applications of steganography and its techniques within different digital media technologies [17-19]. Authors of [17] considered digital watermarking and steganography for digital multimedia over the Internet. They provided a comprehensive overview of the different aspects, mechanisms, and techniques of digital watermarking technology that can be used to guarantee authenticity. However, the work presented in [18] gave an overview of steganography techniques applied to the protection of biometric data. The work of [20] presented steganography method by implement a random key generator, where the Stream cipher (LFSR) was used as the basic idea behind random key generator.

Being said that, although the techniques of cryptography and steganography have been utilized separately, many works have combined them to achieve more secrecy. One can consider in this regard, the works presented in [21-30]. It is notable that the least significant bit has commonly used in these works. For example, the work of [21] introduced a hash-based LSB method for video steganography that conceals secret data or information within a video. In contrast, the work of [22] suggested a way to embed text in video files using LSB substitution. The embedding is done in a location in LSB bits according to equations noted in the paper. The advantage of this method is a simple and successful process for hiding secret messages more securely.

Authors of [23] proposed an approach that combines cryptography and steganography technique to conceal hidden text into a video file in two steps. The text is ciphered using an AES algorithm and then inserted into a video file using LSB substitution. The approach was applied to video frames in 1 LSB, 2 LSBs, and 3 LSBs of each pixel. Results showed that the security was increased. In the same vein, authors of [24] developed a system by combining cryptography and video steganography in to conceal text in a video file. They implemented the proposed system in two layers, where the confidential data are encrypted using an AES algorithm in the first layer, the sensitive data are concealed in a video cover using a motion vector in the second layer. The advantage of this approach is that the hidden data do not distort the video file, hence keeping the quality of the video acceptable. Moreover, authors of [25, 29] followed the same technique. They hided the data using motion vectors on MPEG-2 compressed video. The research experiments were measured according to: video quality and data size. Results showed that even with increased data size, the distortion of the video quality is low.

Another method using many-covers audio and video was presented in [26]. They implemented a method that combines cryptography with audio and video steganography, with the intent of concealing text and images simultaneously inside the audio–video file. They suggested encrypting text and images using an advanced chaotic technique. In addition, a technique using LSB insertion into a video file to hide secret text was proposed in [27], where a data-hiding technique embeds the information based on the stego key generated from polynomial equations. Authors of [28] introduced a method to enhance security in video steganography. They used 4 LSBs substitution to conceal an enormous amount of data in a video in specific frames. Although it is hard to find in which parts of the video the text is embedded, however the security is increased because a human observer cannot detect the data is hidden. One can consider also the work presented in [29], which aimed to improve video steganography by designing a method to embed the text file in a video file differently by utilizing DCT and LSB. The advantage of this system is that data is hidden highly securely and consistently in AVI videos.

The combination of steganography and cryptography was also considered in many works to hide data in images [31-36]. The work of [31-33] combined cryptography and steganography techniques, where the RSA cipher was used for encryption/decryption and least significant bit image

steganography for hiding data. The work of [34] followed the same vein where the RSA encryption technique was used to encrypt data that will inserted into cover image by mapping using breadth first search (BFS). The least significant bit (LSB) technique for cover image and the most significant bit (MSB) was also used as steganography method for secret image in [35], while the authors of [36] presented a two-level securities steganography method. They are using 2D Arnold Cat Map technique to scramble secret data in a random order after that encrypted data is concealed behind a cover image using basic LSB method.

Despite the fact that all the above methods improved video and image steganography by designing methods to embed the text file in a video or image file. They have been investigated and well thought-out to propose our enhanced model. Our method combines cryptography and steganography as two independent sequential levels with all their security features for securing transferable data through an insecure channel. Next sections will present the design and implementation of our model.

III. BACKGROUND MATERIAL

In this section, primary concepts that are related to our approach are firstly presented before discussing the details of our work.

A. Steganography

Steganography could be defined as the covered, hidden or secret writing. This definition could include such different issues as writing with invisible ink or hiding copyright information into a multimedia file (e.g., audio, image, or video) [5]. In fact, there are several applications of steganography and its techniques. One can consider in this regard the works presented in [17-19], where several applications of steganography were developed to secure the transfer of a secret message between authorized entities such as: One-way hash value substitution and Digital watermarking.

In One-way hash value substitution, a variable length input is taken and an output string with a static length is created to verify that the original variable length input has not been changed [37]. However, Digital watermarks, are included or hidden into digital media files that are needed to be fingerprinted for copyright protection. Specifically, Digital marks are hidden within the original files that could be either image, audio or video files, which make them undetectable to public [5].

Steganography has been utilized for transferring a secure message through communication in several works between different parties as they considered the main goal of steganography is to transfer a message and avoid drawing suspicion to its transmission [21-24]. This motivates us to use and apply such steganography application in our model.

It is worth mentioning that the main technique used in steganography is the replacement of the least significant bits, which was described by [38] as the process of hiding secret information by replacing the LSB of pixels of images or samples of audio files with the information to be hidden. To make things clearer regarding this methodology, it could be summarized as replacing all the least significant bits of the host file with the hidden message. Some pixels or audio samples may be left to incorrupt the format of the host file. However, it is suitable and easy for pictures but in the case of audio, the process is harder. This is because the replacement of LSB in 8-bit audio may add noise that is audible, which means that the quality of the audio file will be affected.

In this work, to hide encrypted data, the LSB technique is used whereby Audio wave file is used as a cover file to hide data into it. A text file has also considered to evaluate the system, which has great advantage that it enables a user to hide a file of any type into an audio wave file and Text Format. Moreover, the work also highlights the differences between the original file and the stego file.

B. Cryptography

Cryptography deals with hiding and verification information using different protocols, algorithms and strategies to encrypt/decrypt the sensitive data [14]. It can be divided into two main types or techniques: Symmetric and Asymmetric. While the Symmetric encryption (also called single-key encryption) uses only one key or the same key for the process of encryption and decryption, the Asymmetric encryption (also called public-key encryption) uses two different keys to accomplish the same process [15]. However, a problem then raised in distributing keys, where a trusted party is needed to deliver the keys. Thus, many exchange algorithms have been proposed in the literature. The most significant one was proposed in [39], which is the Diff-Hellman. This algorithm is not used to encrypt data; however, it is used to enable two parties to exchange the key that they want to use to encrypt their messages.

This indicates that there is more than one proposed solution for the same problem. This research is also proposing such a solution for this problem as solving such problem is one of the main aims of the research. The problem can be formulated as the following question:

- *Can the combining of cryptography and steganography solve the key exchange problem?*

This research answers this question, where the secret key is hidden and transferred through an innocent looking file transfer through the using of steganography. The file is then encrypting before hiding it. However, a question then raised of what type of algorithms is needed. Throughout the research, several encryption algorithms were investigated and studied where the One-Time Pad cipher was selected as it handles efficiently the weakness in the substitution and permutation algorithms as discussed in [40].

C. Compression

Data compression has been utilized for transforming text into a special code or run length encoding of image data. However, a modern model paradigm for coding defines data compression as the production of output string from combining an input string with a model [41]. This output is usually a compressed version, smaller in size than the input string and it is called the coded string. The decoder should be able to retrieve the original string from the coded string by accessing the same compression model. In fact, there are two main types for compression, lossless and lossy techniques. These techniques are used to compress text and other discrete data, computer-generated data, and some images and video data types. In lossless technique, if the compression operation is inversed and the output compressed data is decoded, the original data will be retrieved exactly as it was. In contrast, when decompression of data is performed in lossy techniques, the output is data that is similar to the original but not exactly the same. So, the information in the original data and compressed data is not the same. Based on the differences between the two techniques, this work used lossless technique to compress only text files.

Being said that, authors considered the compression process a type of data hiding [41]. Of course, data hiding here is different from what could be considered as steganography. In this work, a discussion and argument are introduced to justify the use of compression throughout this research. The compression algorithm LZ77 [42] is used to compress the text in order to hide a larger data size; justification of this choice is also discussed in the next sections.

IV. SYSTEM DEVELOPMENT AND EVALUATION

Previous sections discuss different issues that lead to developing a software that hides data, either file or text, into a wave audio file. In addition, the discussion also clarifies the required encryption algorithms and the compression algorithm. Since the requirements are fixed at the beginning of the software developmental process, the waterfall developmental model has been chosen to be followed in developing of our system. Coding and testing stages are followed the design stage. In the coding phase, the code was established using Java language. The software development passed through two main iterations each of which passed through different phases of waterfall model, requirements analysis, design, coding and testing. Second iteration had some modifications over the first one. These modifications were extracted and concluded in some stages of developmental cycle. Modifications were done based on either discovering some errors or defects in one of the stages or based on extra information.

A. System Development

The system is built and consisted of the following components. Figure 1 illustrates the package diagram of the system, which shows interactions between its components.

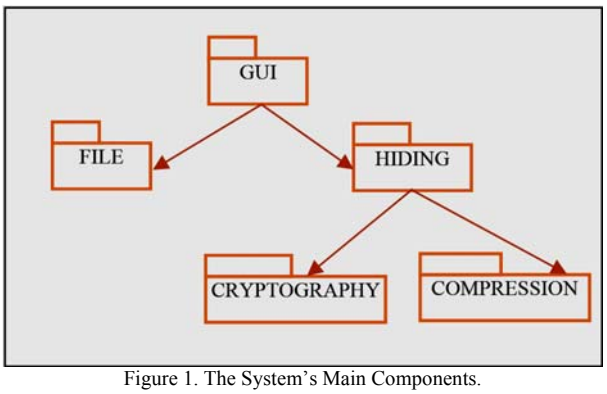

Figure 1. The System's Main Components.

- *GUI package,* which is responsible for showing different graphical user interface components to the user and receiving his/her orders through these components. Additionally, it shows different messages to the user, clarifying for him/her the success of a process or its failure for any reason. Such messages could be a message saying that the selected file has been hidden into the selected wave file.
- *FILE package,* which holds all the functionalities and classes that are responsible for managing different type of files, specifically wave files. It creates and provides other packages with the required information about selected files by the user. Such piece of information could be the size of the selected wave file and the file to be hidden, from which hiding size limits could be calculated.
- *HIDING package,* which provides all the functionalities that are required to hide data into and extract data from a wave file. Many techniques have been studied and investigated to choose the suitable one. For example, Replacement of Least Significant Bits which is a simplest technique for hiding information within a file. It replaces the LSB of pixels of images or samples of audio files with the information to be hidden; Patchwork Algorithm works in different manner. In the case of images, it randomly selects pairs of pixels and makes some changes on them by making the brighter pixel brighter and the darker pixel darker. For audio files, it works by choosing randomly some audio samples and increases their amplitude contrast; Transform Domain Technique, which depends on embedding a mark into the transform domain of the host file. This package uses LSB technique to hide data. It also provides information about number of bits that are used to hide data. It depends on cryptography package because hiding process requires encryption while extracting process requires decryption. It also depends on compression package because hiding text

needs compression before hiding and decompression after extraction.

- *CRYPTOGRAPHY package,* which holds all techniques and functionalities that are required for encryption and decryption process. It does encryption before data hiding and decryption after data extraction. To choose a suitable algorithm to be used in this work, an investigation was carried out to determine that algorithm. Throughout the research, many encryption algorithms were studied. They can be categorized into Substitution, permutation and XORing algorithms. However, many algorithms were eliminated because of their low security [40]. So, several algorithms were between the candidate's algorithms. The DES for example, is a well-0known security and its wide usage. This algorithm was investigated throughout the research. The algorithm basic work is simple as it depends on substitution and permutation repeatedly and each step is called a round. The work of algorithm depends on repeating 16 rounds. Apparently, this needs too long time. Another wide use and high security algorithm is RSA. It gets its high security and power from its long prime keys (128 – 256) bytes. These numbers are multiplied, raised to powers or they could be the power itself. Despite the high security this algorithm provides, however, it also needs too long time. The Hash algorithms were also considered. However, this type of algorithms is considered as a one-way encryption where the original message cannot be retrieved, and the encrypted cipher cannot be decrypted. This is not in the case of this project, so it is also eliminated from the investigation. Thus, two algorithms were selected as candidates (Vigenere and One-Time Pad ciphers) as they handle efficiently the weakness in the substitution and permutation algorithms as discussed in [40]. However, Vigenere has a weakness including use of one key of a specific length to encrypt the whole message, which may lead to break the algorithm. So, One-Time-Pad appeared as an alternative.
- *COMPRESSION package* contains all techniques and functionalities that are required to do compression. For this research purposes, this package contains only a text compression technique, but the design allows extending this package. Through the investigation of a suitable algorithm for compression to be used in this research, several algorithms have been investigated such as: Run Length Encoding, which performs compression depending on the redundancy of characters; Huffman Coding that represents the most common letters (the most redundant letters) in the text by shorter binary code; and LZ-77 Encoding, which is dealing better with text compression than Huffman and arithmetic coding algorithms. LZ-77 influences the academic and forms the basis of several compression schemes, including the DEFLATE algorithm used in PNG and ZIP. It is used to analyse input data and determine how to reduce the size of that input data by replacing redundant information with metadata. Sections of the data that are identical to sections of the data that have been encoded are replaced by a small amount of metadata that indicates how to expand those sections again [42]. Moreover, it does not depend on any type of tables or data structures. This makes it suitable for this research as it needs no more space in the cover file to hide any more information than the compressed message itself. However, to be able to adopt this algorithm, some adaptations are applied to it.

B. System Evaluation

In this section, we will turn our focus to the system's evaluation process. Here, two important questions could be raised regarding the efficiency of the proposed technique, which are:

- *What is the effect of hiding data on the quality of a wave file into which data is hidden (stego file)?*
- *How much data could be hidden using this technique at an acceptable level of sound quality?*

To evaluate the proposed approach, different wave files are chosen and data (after encryption and compression) is hidden into them, using the developed software. Effect of hiding data into different wave files is tested using external software tools and the ideal data size that could be hidden is calculated.

1) Software used: Many softwares have been utilized in this research; however, due to the space limitations, we give brief description for each one.

- *Wave Pad and NCH Tone Generator software ®:* are used to generate different sine wave tones as wave files and compare between original (cover) and stego files. They can be used to compare the amplitude in a time sequence and the amplitude in a frequency sequence [43-44].
- *Note Pad software ®:* is used to generate text files with different sizes. This facilitates testing data size that could be hidden into a specific wave file.
- *Wave Files:* to perform an unbiased experiment, four different wave files that represent different sounds styles are used: file with a soft song notes without any beats, file with a soft song notes and a human voice, file with a song mixture of hard notes and human voice along with beats, and finally, file contains just a human voice without any music. Additionally, four files that contain sounds with specific frequencies are also taken into the sample. These files are generated using a software sine wave generator called NCH Tone Generator. Off course, using such sound samples allows evaluation of the proposed technique over different sound styles.

2) Methods: All sound samples are restricted to a length of three seconds to decrease the CPU time and memory resources required to process the files. Specifying the time

of the first four wave file is done using Wave Pad software. Files specifications are fixed as follows:

Bit Rate = 4411 kbps.
Audio Sample Size = 16 bit.
Channels = 1 channel.
Audio Sample Rate = 44 KHz.
Audio Format = PCM.

The above specifications are chosen because preliminary studies show that such specifications provide good quality audio files and because they are the same specifications advised by [4]. The wave file specifications are fixed beforehand to provide control conditions for the experiment. This also allows the elimination of any differences between different wave files used for the experiments.

3) Data size: To study the efficiency of the proposed technique regarding the amount of data that can be hidden, mathematical rather than subjective evaluation is carried out. Available size to hide data is calculated when 1 bit, 2 bits, 3 bits, 4 bits, 5 bits, 6 bits, 7 bits and 8 bits are used for hiding. Effect of hiding data as file or text and effect of using Pohlig-Hellman or One-Time-Pad algorithms on increasing or decreasing available size for data hiding are justified and discussed. To study the difference between hiding data as a file and hiding it as a text, the use of compression algorithm, LZ77, has been evaluated. Text has been compressed and the number of bytes, which should have been hidden before and after compression process, have been calculated.

V. RESULTS AND DISCUSSION

A. File Quality

Results of the experiment show that for all wave files tested, the distortion as a result of hiding data is not audible when the depth of bit hiding is 1 bit, 2 bits, 3 bits and 4 bits. However, when hiding depth reach 5 bits, an audible hiss can be heard in case of the four sine wave files. Increasing the depth level to 6 bits leads to an audible hiss in all files except the one containing hard notes song that has a hard or strong music. Further increasing the depth to 8 bits level of hiding data does not result in any audible distortion in the sample file. It is important to state that previous results depend on researcher judgment only.

On the other hand, the hard style song appears as an odd result because hiding reaches its eighth bit level and no audible perception change is noticeable. High amplitude and high range of frequencies of such songs may be the reason behind the absence of any audible distortion after hiding data within wave file. This result could not be justified as no research has studied or justified such phenomenon.

It is important to mention here that all the above results and discussion are valid for both Pohlig-Hellman and One-Time-Pad encryption techniques. It must be remembered that the choice of encryption technique used will not affect the quality of the stego file because the encryption methods do not affect how data is hidden within a wave file.

The following Figures (2 - 7) show some resulting data and the effect of hiding data into a wave file. Sine wave of 1000Hz is chosen to present results because it gives the clearest indication of the effect of hiding data within a sound wave file. From the results presented in these, it is clear that there is a difference in the frequency spectrum as a result of hiding data. Although when comparison is done between 2, 3, 4 and 5 bits the difference is not obvious, it becomes obvious when comparing 1 and 8 bits.

From these figures, it could be noticed that all the difference is concentrated, almost, in the right-hand side. It is well known fact that the human ear can only recognise sounds having frequencies between 20 – 20,000 Hz. Therefore, if the distortion in frequencies or the effect of hiding affects frequencies beyond these ranges, this distortion will not be perceived although it is still detectable using sound analysers. This could justify transparency in perception in case of hiding bits in the range of 1 bit to 4 bits. The effect is almost concentrated in the area that cannot be heard by human ear. But in the case of hiding 8 bits, frequency change is closer or in the human ear detectable area.

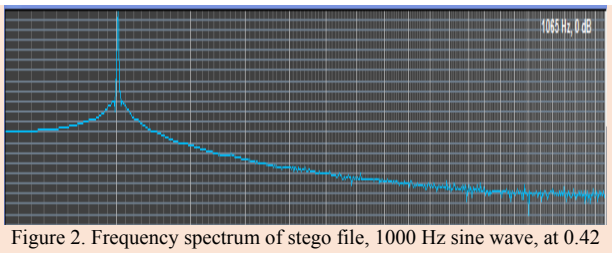

Figure 2. Frequency spectrum of stego file, 1000 Hz sine wave, at 0.42 sec. using 1 bit to hide data.

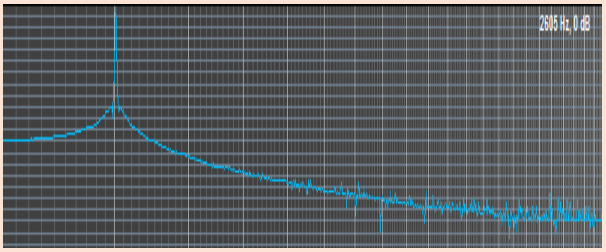

Figure 3. Frequency spectrum of stego file, 1000 Hz sine wave, at 0.42 sec. using 2 bits to hide data.

Figure 4. Frequency spectrum of stego file, 1000 Hz sine wave, at 0.42 sec. using 3 bits to hide data.

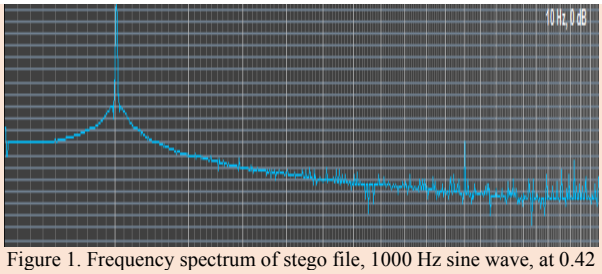

Figure 1. Frequency spectrum of stego file, 1000 Hz sine wave, at 0.42 sec. using 4 bits to hide data.

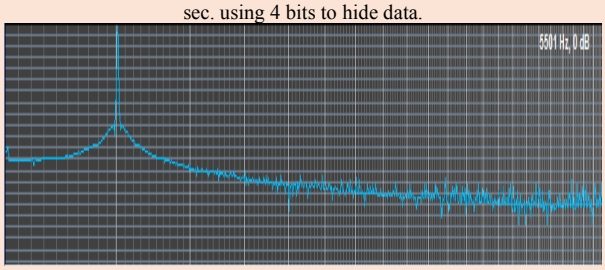

Figure 2. Frequency spectrum of stego file, 1000 Hz sine wave, at 0.42 sec. using 5 bits to hide data.

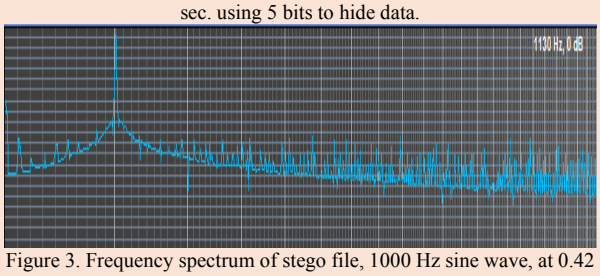

Figure 3. Frequency spectrum of stego file, 1000 Hz sine wave, at 0.42 sec. using 8 bits to hide data.

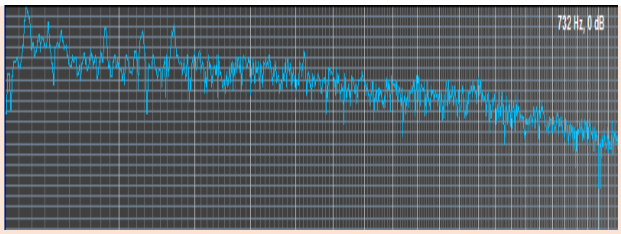

Figure 4. Frequency spectrum of cover file contains hard sound style at 0.63 sec.

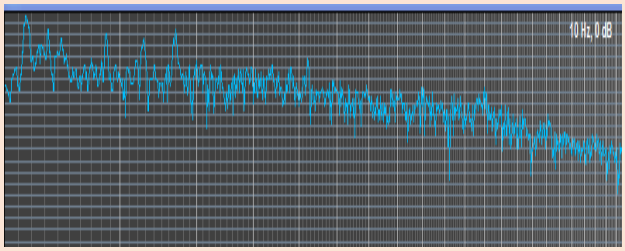

Figure 5. Frequency spectrum of stego file contains hard sound style, at 0.63 sec. using 8 bits to hide data.

B. Data Hidden Size

Although using any of the two encryption techniques does not affect the stego file quality, effect of using either Pohlig-Hellman or One-Time-Pad encryption techniques appears clearly in the amount of data that could be hidden into one wave file. Results show that using Pohlig-Hellman technique reduces data size that could be hidden to almost half, compared to One-Time-Pad technique. In other words, using One-Time-Pad data that could be hidden is double in size of data that is hidden using Pohlig-Hellman. This is because Pohlig-Hellman algorithm depends on choosing big number as modulus. This results into big numbers when data is encrypted. During the development of the software, trials to reduce these numbers to the smallest size resulted into the encryption of each byte by two bytes. So, each byte of a file or text that is to be hidden is encrypted by two bytes of encrypted data. While in One-Time-Pad technique each byte of a file or text to be hidden is encrypted with the same size, one byte, therefore it keeps the size of encrypted data similar to unencrypted one.

Calculating data size that could be hidden into a wave file does not need any experiment. Instead, simple mathematical calculations could lead to better results. Using any of the tested wave file samples, sine wave of 1000Hz wave file of 16 bits/sample and a length of 3 seconds as an example, has a size of 264658 bytes. Subtract bytes reserved for the file header, which are 58 bytes, results 264600 bytes, which equals 2116800 bits or 132300 samples. Thus, the size of data that could be hidden into such file using 2 bits, as an example, for hiding can be calculated as follows.

If data to be hidden is a file using Pohlig-Hellman algorithm for encryption, the following bytes are reserved for extra data other than the file that is to be hidden: first byte (1 byte), array length (4 bytes), file name (unknown length, suppose 8 bytes, 4 bytes for the name, 1 for the dot (.) and 3 bytes for the extension), end of name mark (1 byte), p (2 bytes), d (2 bytes). The total is almost, depending on file to be hidden name, 18 bytes or 144 bits. This number should be divided by 2 because 2 bits are used in hiding data which results in 72 samples that are required to hide these extra bits. Subtracting this number from the total number of available samples, results in 132228 samples which are now available to hide the file. Multiplying this number by 2, because 2 bits are used to hide data in each sample, results in 264456 bits or 33057 bytes.

Since Pohlig-Hellman algorithm encrypts each byte with 2 bytes, the number will be reduced to the half, which results in 16528 bytes or almost 16 KB. These bytes are the length of file that could be hidden into a wave file with the above specifications and length.

However, If data to be hidden is a file using One-Time-Pad algorithm for encryption, the following bytes are reserved for extra data other than the file that is to be hidden: first byte (1 byte), array length (4 bytes), file name (unknown length, suppose 8 bytes, 4 bytes for the name, 1 for the dote (.) and 3 bytes for the extension), end of name mark (1 byte), encryption keys (unknown, suppose at most 24 as calculated above). The total is almost 38 bytes or 304 bits, depending on file to be hidden, name and encryption keys lengths. Applying the same calculations above result 264296 bits or 33037 bytes available to hide data. Because One-Time-Pad algorithm encrypts each byte with one byte, the available

space to hide data is 33037 bytes or almost 32 KB file size, which is double the size that Pohlig-Hellman algorithm can offer.

The following table summarizes data of available spaces in bytes for data hiding into a wave file using both encryption techniques and different number of bits for hiding and shown in Figure 10.

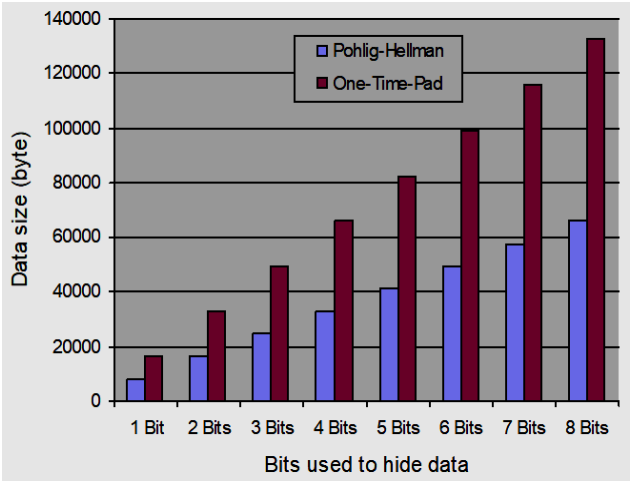

Figure 10. Data size (in bytes) that could be hidden into a wave file of 16 bits per sample format and a size of 264658 bytes.

TABLE I. DATA SIZE (IN BYTES) THAT COULD BE HIDDEN INTO A WAVE FILE OF 16 BITS PER SAMPLE FORMAT AND A SIZE OF 264658 BYTES.

	Pohlig-Hellman	**One-Time-Pad**
1 Bit	8259	16499
2 Bits	16528	33037
3 Bits	24797	49574
4 Bits	33066	66112
5 Bits	41335	82650
6 Bits	49603	99187
7 Bits	57872	115724
8 Bits	66141	132262

From the above table and the given graph, it is clear that the available size for hiding data is increasing as the number of bits to hide data is increased. It also clarifies that One-Time-Pad provides a space twice as large as that is provided by Pohlig-Hellman.

Ultimately, if a text is to be hidden instead of a wave file, the same calculations are valid for both encryption techniques except bytes that are reserved for file name and its mark (according to the above assumption, they are 9 bytes). However, when a text is chosen to be hidden, a compression algorithm, LZ77, is used to compress the text before hiding it. Results of using this algorithm show that it works perfectly if there are similar characters in the introduced message. This is because it compresses any five or more similar sequences of characters in the text.

Evaluating of such algorithm is difficult because its efficiency in compressing a text depends on the text itself. For example: consider words like “the and that”. the first takes five characters (three characters, one space before it and one after it), while the second takes six characters. Each of them will be compressed and sampled in the compressed text as four characters only. This leads to reducing the number of bytes needed to be hidden from five and six, respectively, to only four bytes in each case. Imaging how many times these words are included into any normal text or message could give an idea of the benefit of this compression algorithm in reducing size of data to be hidden. To evaluate this algorithm, an original text of 1470 characters has been compressed and produced a text of 1379. The difference is 91 characters or 91 byes that are not needed to be hidden any more.

It is important to state that number of bits per sample in a wave file does not affect size of data that could be hidden. So, the above numbers are valid even if 8 bits/sample wave file is used for hiding data instead of a wave file of 16 bits/sample although a wave file size of 16 bits/sample itself is twice the size of a wave file of 8 bits/sample. There is no difference between wave files of 8 bits/sample and 16 bits/sample because data is hidden into samples regardless of whether these samples are 8 bits or 16 bits. The only difference between wave files of 8 bits/sample and those of 16 bits/sample appears in stego file quality as discussed before.

VI. CONCLUSIONS AND FUTURE WORKS

In this research, we have investigated the impact of combining different security measures on transmitted data. To do this, a system was developed specifically for this purpose. In addition, two studies have been conducted, the first one was developed to determine if there are any differences between stego and original files using different wave files of 16 bits per sample of different sound styles. Results demonstrated that hiding data using up to 4 bits does not show any difference between these files. Otherwise, the hidden data could be easily detected and discovered. Another main factor that might be focused on is the text files. The second experiment considers this issue and was conducted on text files using the same calculations for both encryption techniques. Moreover, a compression algorithm (LZ77) is used to compress the text before hiding it. Results show that it works perfectly in this context.

As a future work, we will use the research described here as a foundation for developing effective securing systems. In addition, we are planning to consider other encryption algorithms to generalize our results. Another main factor that might be considered is the host file or the digital media that will be used as a carrier file into which the data or the message will be hidden. Different carrier file types will

definitely have different algorithms and techniques to hide data into them and different data rate they can hold. Therefore, further validations are desperately required. That being said, this is a large topic and there are numerous opportunities for additional research that would significantly extend the functionality of the current research.

ACKNOWLEDGMENT

The authors are grateful to the Applied Science Private University, Amman-Jordan, for the full financial support granted to cover the publication fee of this research article.

REFERENCES

[1] Hashem, Ibrahim Abaker Targio and Yaqoob, Ibrar and Anuar, Nor Badrul and Mokhtar, Salimah and Gani, Abdullah and Khan, Samee Ullah, "The rise of "big data" on cloud computing: Review and open research issues," Information systems, vol. 47, 2015, pp. 98-115.

[2] Altamimi, Ahmad, "SecFHIR: A security specification model for Fast Healthcare Interoperability Resources," International Journal of Advanced Computer Science and Applications (ijacsa), vo. 7(6), 2016.

[3] Lindell, Yehuda and Katz, Jonathan, Introduction to modern cryptography, Chapman and Hall/CRC, 2014.

[4] Cole, E., 2003. Hiding in plain sight: Steganography and the art of covert communication. Indianapolis: Wiley.

[5] Shih, Frank Y, Digital watermarking and steganography: fundamentals and techniques, CRC press, 2017.

[6] Sajjad, Muhammad and Muhammad, Khan and Baik, Sung Wook and Rho, Seungmin and Jan, Zahoor and Yeo, Sang-Soo and Mehmood, Irfan, "Mobile-cloud assisted framework for selective encryption of medical images with steganography for resource-constrained devices," Multimedia Tools and Applications, vol. 76, No. 3, 2017, pp. 3519-3536.

[7] Mishra, Rina and Bhanodiya, Praveen, "A review on steganography and cryptography," Proc. 2015 International Conference on Advances in Computer Engineering and Applications (ICACEA), IEEE Press, 2015, pp. 119-122.

[8] Saxena, Aumreesh Kumar and Sinha, Sitesh and Shukla, Piyush, "Design and Development of Image Security Technique by Using Cryptography and Steganography: A Combine Approach," International Journal of Image, Graphics & Signal Processing, vol. 10, No. 4, 2018.

[9] Chauhan, Shivani and Kumar, Janmejai and Doegar, Amit and others, "Multiple layer text security using variable block size cryptography and image steganography," Proc. 3rd International Conference on Computational Intelligence & Communication Technology (CICT), IEEE Press, 2017, pp. 1-7.

[10] Zhou, Xinyi and Gong, Wei and Fu, WenLong and Jin, LianJing, "An improved method for LSB based color image steganography combined with cryptography," Proc. 15th International Conference on Computer and Information Science (ICIS), IEEE Press, 2016, pp. 1-4.

[11] Gencogu, Muharrem Tuncay, "Combining Cryptography with Steganography," ITM Web of Conferences, Vol. 13, 2017, pp. 01010.

[12] Bojinov, Hristo and Subramanian, Ananthan, Network storage server with integrated encryption, compression and deduplication capability, Google Patents, 2016.

[13] S. Usha, G. A. Sathish Kumal, K. Boop athybagan., A. Secure Triple Level Encryption Method Using Cryptography and Steganography. International Conference on Computer Science and Network Technology, IEEE. (2011)

[14] Thambiraja, E and Ramesh, G and Umarani, Dr R, "A survey on various most common encryption techniques," International journal of advanced research in computer science and software engineering, vol. 2, No, 7, 2012.

[15] Agrawal, Monika and Mishra, Pradeep, "A comparative survey on symmetric key encryption techniques," International Journal on Computer Science and Engineering, vol. 4, No. 5, 2012, pp. 877.

[16] Pakshwar, Rinki and Trivedi, Vijay Kumar and Richhariya, Vineet, "A survey on different image encryption and decryption techniques," International journal of computer science and information technologies, vol. 4, No. 1, 2013, pp. 113-116}.

[17] Shih, Frank Y, Digital watermarking and steganography: fundamentals and techniques, CRC press, 2017.

[18] Douglas, Mandy and Bailey, Karen and Leeney, Mark and Curran, Kevin, "An overview of steganography techniques applied to the protection of biometric data," Multimedia Tools and Applications, vol. 77, No. 13, 2018, pp. 17333-17373.

[19] Rani, Meenu and Bedi, Charandeep Singh, "Review of Various Image Steganography Techniques and Different Type For Data Hiding Scheme," ZENITH International Journal of Multidisciplinary Research, vol. 5, No. 9, 2015, pp. 19-23.

[20] Ismael Abdul Sattar; Methaq Talib Gaata Image steganography technique based on adaptive random key generator with suitable cover selection," Porc. Annual Conference on New Trends in Information & Communications Technology Applications (NTICT), 2017, pp. 208-212.

[21] P. R. Deshmukh, & B. Rahangdale, "Hash based least significant bit techniquefor video steganography", International Journal of Engineering Research and Applications, Vol 4, No. 1, January 2014, pp. 44-49.

[22] K. U. Singh, "Video-Steganography: Text Hiding in Video by LSB Substitution", International Journal of Engineering Research and Applications, Vol. 4, No. 5, May 2014, pp. 105-108.

[23] H. Gupta, & S. Chaturvedi, "Video Steganography through LSB Based Hybrid Approach", International Journal of Engineering Research and Development, Vol. 6, No. 12, May 2013, pp. 32-42.

[24] N. Prabhakaran, & D. Shanthi, "A New Cryptic Steganographic Approach using Video Steganography", International Journal of Computer Applications, Vol. 49, No. 7, July 2012, pp.32-42.

[25] H. Aly, "Data Hiding in Motion Vectors of Compressed Video Based on Their Associated Prediction Error", IEEE Transactions on Information Forensics and Security, Vol. 6, No. 1, March 2011, pp.14-18.

[26] P. Praveen, & R. Arun, "Audio-video Crypto Steganography using LSB substitution and advanced chaotic algorithm", International Journal of Engineering Inventions, Vol. 4, No. 2, August 2014, pp. 01-07.

[27] A. Swathi, & S. A. K. Jilani, "VideoSteganography by LSB Substitution Using Different Polynomial Equations", Proc. 18 International Journal of Computational Engineering Research, Vol. 2, No. 5, September 2012, pp. 1620 - 1623.

[28] S. K. Moon, & R. D. Raut, "Analysis of secured video Steganography using computer forensics techniques for enhances data security", Proc. IEEE Second International Conference on Image Information Processing (ICIIP), 2013, pp. 660 - 665.

[29] V. Bodhak, & L. Gunjal, "Improved protection in video Steganography using DCT & LSB", International Journal of Engineering and Innovative Technology (IJEIT), Vol. 1, No. 4, 2012, pp. 31 - 37.

[30] Shivani Chauhan; Jyotsna; Janmejai Kumar; Amit Doegar "Multiple Layer Text security Using variable block size cryptography and image steganography," Proc. 3rd International Conference on Computational Intelligence & Communication Technology (CICT), Page(s):1-7, India2017

[31] Shivani Chauhan; Jyotsna; Janmejai Kumar; Amit Doegar "Multiple Layer Text security Using variable block size cryptography and image steganography," Proc. 3rd International Conference on Computational Intelligence & Communication Technology (CICT), Page(s):1-7, India 2017

[32] Shubhi Mittal; Shivika Arora; Rachna Jain "PData security using RSA encryption combined with image steganography," Proc. 1st

India International Conference on Information Processing (IICIP), Page(s):1-5, India-2016

[33] M. Saritha; Vishwanath M. Khadabadi; M. Sushravya "Image and text steganography with cryptography using MATLAB," Proc. International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES) page(s): 584-587, India-2016.

[34] Mamta Jain; Rishabh Charan Choudhary; Anil Kumar "Secure medical image steganography with RSA cryptography usingdecision tree," Proc. 2nd International Conference on Contemporary Computing and Informatics (IC3I), Page(s):291-295, India- 2016

[35] Nikhil Patel; Shweta Meena "LSB based Image steganography using Dynamic key cryptography" 2016 International Conference on Emerging Trends in Communication Technologies (ETCT), Pages(s): 1- 5, India-2016

[36] Rupali Bhardwaj; Divya Khanna Enhanced the security of image steganography through image encryption," Proc. Annual IEEE Conference (INDICON), Page(s):1-4, India-2015

[37] Kessler, Gary C and Hosmer, Chet, "An overview of steganography," Advances in Computers, vol. 83, 2011, pp. 51-107.

[38] Thangadurai, K and Devi, G Sudha, "An analysis of LSB based image steganography techniques," Proc. International Conference on Computer Communication and Informatics (ICCCI), IEEE Pross, 2014, pp. 1-4.

[39] Boyko, Victor and MacKenzie, Philip and Patel, Sarvar, "Provably secure password-authenticated key exchange using Diffie-Hellman," Proc. International Conference on the Theory and Applications of Cryptographic Techniques, Springer Press, 2000, pp. 156-171.

[40] Smart, Nigel Paul and others, Cryptography: an introduction, vol. 3, McGraw-Hill New York, 2003.

[41] Sayood, Khalid, Introduction to data compression, Morgan Kaufmann, 2017.

[42] Wolff, Francis G and Papachristou, Chris, "Multiscan-based test compression and hardware decompression using LZ77, Proc. International Test Conference, IEEE Press, 2002, pp. 331-339.

[43] WavePad Audio Editing Software. https://www.nch.com.au/wavepad/index.html

[44] NCH Tone Generator. https://www.nch.com.au/tonegen/index.html.