Desarrollo de software esteganográfico con criptográfia asociada

Jorge Eduardo Rivadeneira Muñoz, Basel Halak

Escuela de Electrónica y Computación, Universidad de Southampton, University Road, Southampton, Reino Unido, SO17 1BJ.

Autores para correspondencia: jerm1n14@southamptonalumni.ac.uk, bh9@ecs.soton.ac.uk

Fecha de recepción: 2 de mayo 2017 - Fecha de aceptación: 5 de agosto 2017

RESUMEN

Proteger la información es un requisito fundamental para preservar su confidencialidad e integridad. El uso de criptografía y esteganografía, más que un simple compendio de técnicas y algoritmos, se ha convertido en una necesidad urgente en todos los campos en los que la información es un activo valioso. Desafortunadamente, con la mejora de los sistemas informáticos, intrusos y adversarios realizan ataques más rápidos y aplican mejores técnicas de análisis esteganográfico para obtener la información embebida destinada a ser exclusiva entre el emisor y el receptor. A pesar de la existencia de un grupo de técnicas esteganográficas consideradas eficientes, existe la necesidad imperiosa de complementar esos procedimientos con métodos adicionales para mejorar la seguridad de la información incorporada antes del almacenamiento o la transmisión a través de la red pública. Basado en este hecho, el objetivo de este artículo es explicar la propuesta de un software esteganográfico con criptografía asociada, fácil de usar, capaz de encriptar mensajes de texto y embeberlos en un archivo MP3. La arquitectura del sistema consta de dos partes principales, en la primera etapa se implementa un algoritmo de cifrado simétrico, mientras que la segunda fase toma el resultado de la fase de cifrado y la integra en un archivo de música, sin afectar su calidad de audio.

Palabras clave: Encripción, EST, ID3v2, MP3, esteganografía.

ABSTRACT

Data protection is a fundamental requirement to preserve confidentiality and integrity of information. The use of cryptography and steganography, more than just a compendium of techniques and algorithms, has become a pressing need in all fields where the information is an asset. Unfortunately, with the improvement of computer systems, intruders and adversaries perform faster attacks and apply better steganalysis techniques to obtain the embed information meant to be exclusive between the sender and receiver. Despite the existence of a group of efficient steganographic techniques, there is the imperative need to complement those procedures with additional methods to enhance the security of the embedded information before storage or at the time being sent through a public network. This paper presents a user-friendly Security-Enhanced Steganography Software, able to encrypt text messages and embed them in an MP3 file. The architecture of the system consists of two main parts, in the first stage, a symmetric encryption algorithm, mostly used today, is implemented, while the second stage takes the outcome of the encryption phase and embeds it into a music file, without affecting its audio quality, therefore, the modified file can be played on any music player or smartphone that supports this format without any inconvenience.

Keywords: Encryption, EST, ID3v2, MP3, steganography.

1. INTRODUCCIÓN

El almacenamiento y el intercambio de información se consideran como dos de las principales actividades entre los usuarios de Internet, millones de archivos son enviados y recibidos a través de esta red pública poco confiable donde en la mayoría de los casos los datos son susceptibles de ser interceptados por terceros no permitidos. Si los mensajes que enviamos a la red pública no se tratan correctamente utilizando técnicas de seguridad, se podría revelar información confidencial, comprometiendo tanto al remitente como al receptor.

La esteganografía y criptografía han desempeñado un papel esencial en el ámbito de la seguridad. Cada una tiene un enfoque diferente, pero ambas alcanzan el mismo objetivo, que es mantener la información confidencial. Por un lado, la criptografía preserva el secreto de la información en el almacenamiento, o durante la transmisión, mediante la aplicación de codificación y cifrado, mientras que la esteganografía utiliza técnicas de ocultamiento de información dentro de otros objetos (archivos) para pasar desapercibida. Ambos métodos son válidos y contribuyen significativamente en el área de seguridad, sin embargo, en muchos escenarios estos procedimientos actúan de manera independiente. La razón más común de esto es que la esteganografía depende tanto del mensaje que se va a ocultar como del objeto de cubierta utilizado para embeber la información. Por otro lado, las técnicas de cifrado son más genéricas, para ser aplicadas a diferentes tipos de archivos o mensajes. El hecho de que la criptografía sea ampliamente utilizada, especialmente en aplicaciones de software, ha llevado al desarrollo de estándares robustos como el *Advanced Encryption Standard* (AES). Desafortunadamente, la esteganografía no ha tenido la misma buena fortuna, en este campo, hay técnicas limitadas y específicas, dependiendo del mensaje a incrustar y el objeto de cubierta utilizado. Esto ha generado la aparición de adversarios que tratan de socavar estos métodos a través del esteganálisis.

El progreso de la tecnología permite a los esteganalistas, así como a los adversarios, realizar análisis y ataques contra objetos de cobertura en menos tiempo para encontrar información incorporada, a través del uso de hardware y software especializados. Esta es la razón principal por la cual la esteganografía tiene que estar en desarrollo continuo. Hay técnicas esteganográficas que permiten que cadenas de texto o mensajes se incluyan dentro de imágenes digitales y archivos de audio, sin embargo, aplicando diferentes tipos de métodos, el mensaje oculto podría ser recuperado, revelando información secreta. Este problema podría resolverse aplicando algoritmos de cifrado antes del proceso de embebido. Hoy en día existen soluciones de software que incorporan información en imágenes o archivos de audio sin realizar una etapa de encriptación o utilizando reordenamientos muy simples de caracteres, lo que pone en peligro la confidencialidad e integridad de la información. En algunos casos, se desconoce la técnica de cifrado implementada en las herramientas. En otras palabras, su resistencia se basa en la seguridad por obscuridad, violando el principio de Kerckhoffs. (Delfs & Knebl, 2001)

2. ANTECEDENTES Y TRABAJOS PREVIOS

2.1. Esteganografía digital

La esteganografía puede definirse como el arte de esconder información en algún otro elemento que pasa desapercibido y no llama la atención a un tercero dentro de una comunicación entre un emisor y un receptor (Maleki, Jalali & Jahan, 2014). El origen exacto de la esteganografía es desconocido, sin embargo, se puede suponer que comenzó el momento en que surgió la urgente necesidad de transmitir mensajes de forma encubierta. Un sistema esteganográfico se compone de: un mensaje o secreto, un objeto de cubierta, una función o técnica de esteganografía, un canal inseguro, una clave esteganográfico y un objeto esteganográfico. La esteganografía multimedia o digital reúne un conjunto de técnicas en las que un secreto, que se transmite o almacena, se inserta en un objeto de cubierta (texto, música, video o imagen) sin modificar su contenido multimedia y pasar desapercibido a través de una red de comunicación insegura o un sitio de almacenamiento (Anderson, 2008). En ambos escenarios, el objeto esteganográfico podría ser interceptado por un atacante, pero no podrá identificar la alteración del

archivo original utilizado como portador del secreto, aunque si el adversario puede descubrir el cambio todavía no será capaz de entender el mensaje embebido (Petitcolas, Anderson & Kuhn, 1999).

2.2. MP3 como objeto de cubierta

MP3 es un formato de codificación de audio definido en ISO 11172-3 y 13818-3, se basa en una compresión de datos con pérdidas para disminuir el tamaño de un archivo de audio. Si se compara el tamaño de un archivo de audio de CD con el tamaño de archivo MP3, se puede encontrar que el tamaño del archivo MP3 es al menos diez veces menor que el otro, sin afectar la calidad de la música debido a que el oído humano no puede distinguir una diferencia en tasas de bits mayores de 128 kb/s. Esto lo ha llevado a ser considerado como uno de los formatos de codificación más populares y uno de los más utilizados en todo el mundo (Egidi & Furini, 2005). Los archivos MP3 se crean de forma óptima respecto al tamaño, lo que no los hace los mejores objetos de tapa para aplicaciones de esteganografía, sin embargo, dentro de su estructura, tienen campos que pueden ser alterados o incluso llenados con información sin afectar el contenido audio. El archivo MP3 está compuesto por un grupo de *tramas* y *tags*. Cada trama tiene una cabecera y un *payload* donde se almacenan los metadatos. Inicialmente, un archivo MP3 contenía una sola etiqueta (*tag*) llamada ID3v1, localizada en la sección final del archivo, sin embargo, actualmente presentan una etiqueta adicional situada antes del conjunto de *tramas*, llamada ID3v2 (Nilsson & Sundström, 2003).

2.3. Trabajos previos

Modelo de tres capas para la esteganografía de audio

Asad, Gilani & Khalid (2012) han propuesto un modelo basado en un sistema de tres etapas. El secreto pasa por una fase de codificación, para su transformación de caracteres a bits, seguido de un cifrado simétrico que proporciona una capa adicional de seguridad. Para esta propuesta se utilizó AES-256 con un tamaño de bloque de 128 bits. Finalmente, la etapa de esteganografía utiliza una variación del método LSB dependiendo del primer, segundo y tercer bits. Según los autores, esta técnica aprovecha los tres bits menos significativos de cada muestra, que se pueden cambiar sin aumentar la cantidad de ruido en el objeto de cubierta.

Programa de aplicación de esteganografía utilizando un archivo MP3 en un teléfono móvil.

Salman & Kanigoro (2014) han desarrollado una aplicación de *Android* capaz de embeber una clave y un mensaje tanto en la etiqueta ID3v2 como en las tramas de un archivo MP3. Antes del proceso de esteganografía, el mensaje se cifra utilizando el cifrador *McEllice*. El mensaje cifrado se convierte en *bytes* y se oculta entre las tramas del archivo MP3. La clave pública también se embebe, pero en la portada de la figura frontal del archivo que se almacena en la etiqueta. Según los autores, la clave y el mensaje encriptado podrían incrustarse en la portada del álbum, sin embargo, esto restringiría el tamaño del mensaje. En este desarrollo no se especifica qué campos de las tramas han sido modificados para almacenar el mensaje. Para el proceso de recuperación y descifrado, el primer paso consiste en leer la etiqueta, si no se encuentra la clave, el proceso se interrumpe. De lo contrario, el mensaje se extrae, descifra y se muestra al usuario.

Esteganografía usando la portada del álbum de un archivo MP3

Utilizando C#, Sing & Kumar (2016) propusieron y desarrollaron una herramienta de esteganografía que embebe el mensaje secreto en la portada del álbum de un archivo MP3. La diferencia entre el trabajo anterior y este es que el mensaje está en texto plano. Para la parte de esteganografía, se ha utilizado un método de codificación-tiempo.

3. HERRAMIENTA ENCRIPTO-ESTEGANOGRÁFICA (EST)

La primera versión de EST es una aplicación de escritorio desarrollada en VB.NET capaz de cifrar y embeber un secreto (mensaje de texto) dentro de una trama del archivo de audio MP3. Esta aplicación sigue un ciclo de vida de desarrollo de software (SDLC), definiendo requisitos, diseñando la arquitectura, implementando la aplicación, verificando su funcionamiento y rendimiento, pruebas, control de errores y optimización (Futcher & von Solms, 2008). El ciclo de desarrollo elegido siguió un modelo iterativo, donde el análisis de requisitos y diseño se realizó en las primeras etapas del proyecto, mientras que la codificación y la prueba se ejecutaron periódicamente. Como se mencionó anteriormente, la aplicación contiene dos partes principales, el criptosistema y el segmento esteganográfico, cada una proporciona un conjunto de funciones que fueron codificadas y probadas por separado.

3.1. Requerimientos principales

La etapa de cifrado debe estar compuesta por un método de encripción simétrico bien definido (estándar) como el núcleo, mientras que la etapa de esteganografía debe utilizar al menos una de las técnicas esteganográficas conocidas. El objeto esteganográfico debe ser audible, no distorsionado, y sin un aumento significativo en el tamaño del nuevo archivo en comparación con el objeto de cubierta.

La aplicación requiere de dos interfaces gráficas de usuario, la primera se encarga de cifrar y embeber un mensaje en un archivo de audio MP3, mientras que la segunda permite recuperar el mensaje, extrayéndolo y descifrándolo, la Figura 1 detalla los casos de uso de la aplicación.

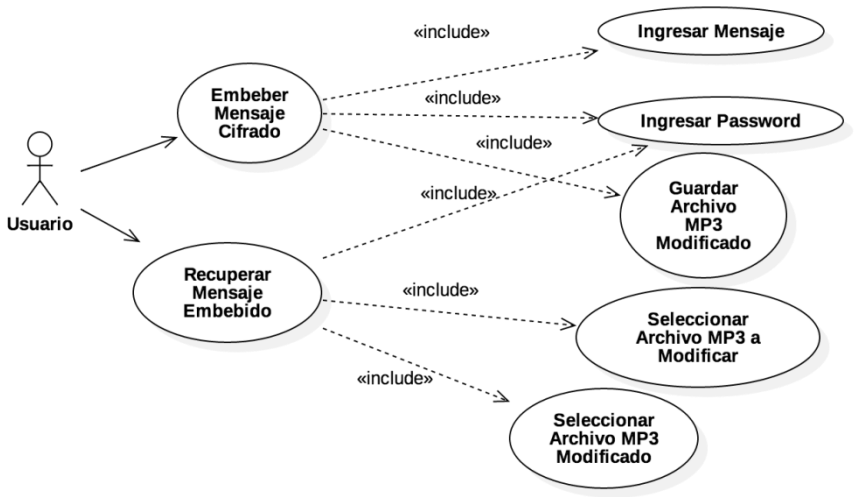


Figura 1. Diagrama de casos de uso de la aplicación.

3.2. Diseño

Para el sistema criptográfico se han acordado las siguientes características: una encriptación y desencripción simétricas fuertes, un texto claro (mensaje), un texto cifrado (el secreto a ocultarse) y una clave robusta de 256 bits (Mathur & Kesarwani, 2013; Daemen & Rijmen, 2013). Con el fin de optimizar el número de bytes que se va a cifrar, se consideró la adición de una etapa de compresión de texto como primer bloque del sistema.

Para la parte esteganográfica, el resultado de la primera etapa se toma como argumento de entrada junto con el archivo MP3. El campo de los archivos multimedia que se cambia son las etiquetas, específicamente la ID3v2, con el fin de no comprometer la integridad de las tramas y afectar a los datos audibles. La imagen del álbum son metadatos almacenada en una trama de la etiqueta ID3v2, elemento en el que se va a implementar la técnica esteganográfica. Dado que el contenido de este campo es una imagen (*True Color*-24bpp), y el tamaño del texto encriptado es pequeño, hemos considerado la aplicación del método de inclusión de bit menos significativo (Atoum, 2015; Bazyar & Sudirman, 2014). Una vez que el objeto de cubierta ha sido modificado, ocultando el mensaje cifrado, se inyecta la trama

específica de la etiqueta ID3v2, y se genera el nuevo archivo MP3. Para el proceso de recuperación, la aplicación debe ser capaz de leer la etiqueta, recuperar la imagen de portada y extraer el mensaje cifrado de ella leyendo el bit modificado de los pixeles, después de obtener el mensaje cifrado el siguiente paso es desencriptarlo usando la contraseña correcta y finalmente la etapa de descompresión.

3.3. Implementación

Como en la etapa de diseño, la fase de implementación se dividió en dos sub-partes, la primera corresponde a la compresión y cifrado, mientras que la segunda parte cubre el análisis del *tag*, la extracción de la imagen del álbum y el método esteganográfico.

Módulo de Compresión y Encripción

Para el módulo de compresión/descompresión se emplea el algoritmo de Huffman a través de la clase *HuffmanCoding* la cual contiene dos métodos principales: *CompressByteArray* y *DecompressByteArray*, tomando como argumento un arreglo de bytes. El mensaje introducido por el usuario de la aplicación es codificado mediante UNICODE, permitiendo caracteres que en ASCII no se definen, como acentos y signos de puntuación especiales que se utilizan en otros idiomas pero no en el idioma inglés. El módulo de cifrado/descifrado se basa en AES-256, a través de la clase *AesCryptoServiceProvider*. Se definió un modo de cifrado CBC (Cipher Block Chaining) con un vector de inicialización y un tamaño de bloque de 128 bits para ambos (Paar & Pelzl, 2012). La derivación de claves se realiza mediante el procedimiento *PBKDF2*, definido en la clase *Rfc2898DeriveBytes* con parámetros como la contraseña de usuario, un valor de *salt* de 256 bits y un contador de iteración (Figura 2).

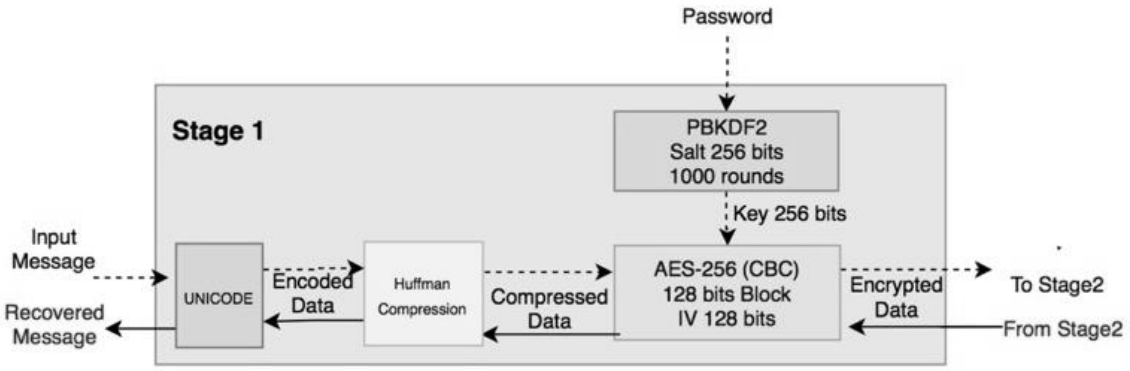


Figura 2. Implementación primera etapa.

Módulo Esteganográfico

El módulo esteganográfico toma los bytes cifrados y un mapa de bits del arte del álbum MP3 como argumentos de entrada. La imagen contenida en el archivo de audio se extrae utilizando métodos específicos de la clase *AttachedPictureFrame*, perteneciente a la biblioteca *taglib-sharp.dll*. En el momento de la implementación, se considera un reacondicionamiento de la imagen extraída, redimensionándola a 300 x 300 píxeles de color verdadero. Para la nueva imagen redimensionada se calcula el número máximo de bytes ζ que se pueden embeber utilizando el método de bit menos significativo:

$$\zeta = \frac{h_{px}.w_{px}.\delta_{pp}}{64} \quad (1)$$

h_{px} y w_{px} son las dimensiones de la imagen en pixeles mientras que δ_{pp} es la profundidad de color en bits por pixel. Por lo tanto, para las anteriores condiciones se tiene que:

$$\zeta = \frac{300\ x\ 300\ x\ 24}{64} = 33750\ [bytes] \quad (2)$$

Después de cambiar el tamaño, la imagen se transforma en un mapa de bits para realizar la técnica esteganográfica. Una vez que los bits han sido modificados, el campo que contiene el nuevo mapa de bits se reemplaza, si la imagen original es mayor que el tamaño del mapa de bits, el campo no disminuye y, por lo tanto, el tamaño del archivo MP3 no cambia. Si la imagen original es más pequeña que el mapa de bits, el campo del contenedor se redimensiona para ajustarse al nuevo BMP (Figura 3).

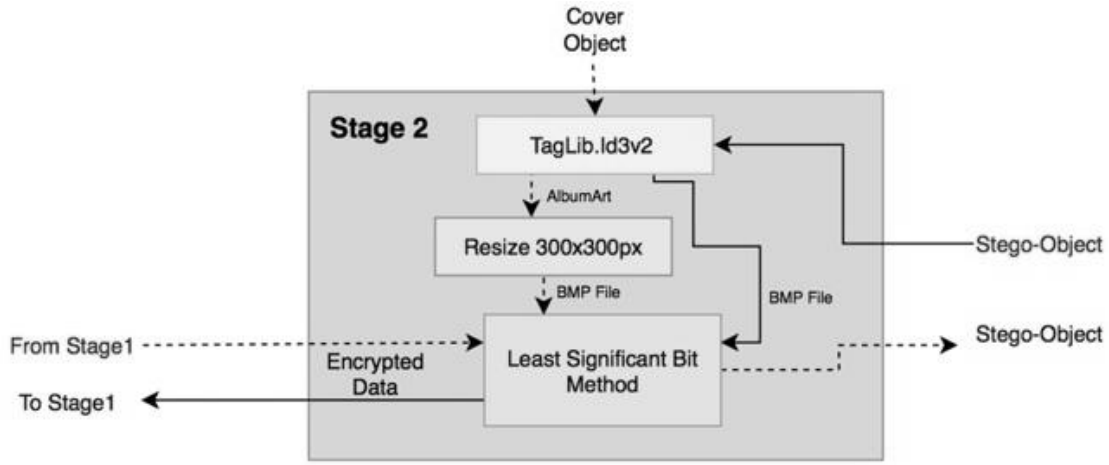


Figura 3. Implementación segunda etapa.

4. RESULTADOS Y DISCUSIÓN

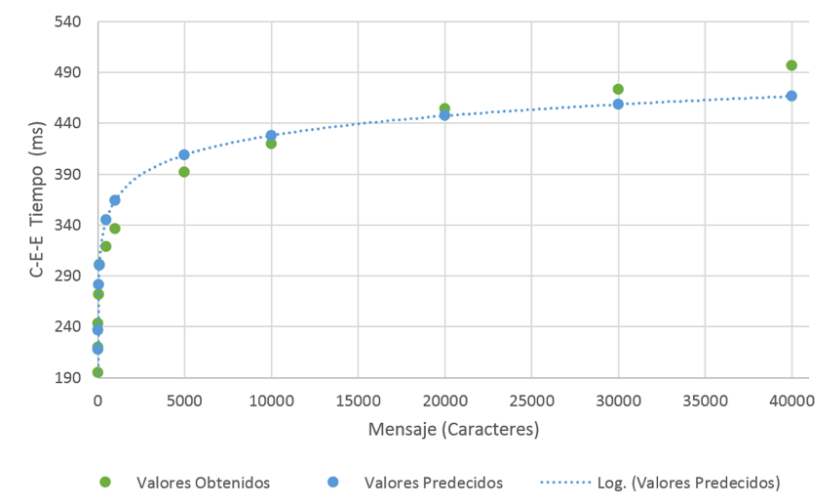
A través del desarrollo de la aplicación, se realizaron algunas pruebas unitarias durante el proceso de depuración. Cuando todas las etapas se fusionaron, y la aplicación pasó de la fase de desarrollo, se realizó una ronda final de pruebas. La última ronda consistió en seleccionar cinco archivos MP3 aleatorios e insertar mensajes de diferentes tamaños, comenzando con 10 caracteres hasta 40000, que era el máximo establecido en la aplicación dada por la limitación explicada en base al resultado de la ecuación 2. Tanto la longitud, como el tamaño de los cinco archivos MP3 utilizados como objetos de cubierta se han grabado antes del proceso de embebido, así como el tamaño y la duración después del proceso. La información se resume en la Tabla 1, incluyendo el número de bytes que se obtuvieron después de las etapas de compresión y encriptación.

Dentro de los resultados, es posible enfatizar que después del proceso de embebido la duración de la pista musical no es diferente de la original. El tamaño del archivo aumenta alrededor de 270000 bytes porque una imagen BMP generada a partir de un mapa de bits 300x300 contiene 27054 bytes, 27000 bytes de datos más 54 bytes de la cabecera BMP. En el caso del segundo archivo, no hay incremento, esto se debe a que la imagen original contenida dentro de su etiqueta ID3v2 es mayor que el tamaño de la BMP generada después de la técnica esteganográfica. Cabe mencionar que el aumento de tamaño del objeto de cubierta es fijo y no depende en ningún momento de la longitud del mensaje. El tiempo de procesamiento también se midió, tanto en el primer proceso, como en el segundo, la Figura 4a muestra el tiempo (ms) requerido por la aplicación para realizar la compresión, encriptación y proceso embebido, mostrando una tendencia logarítmica.

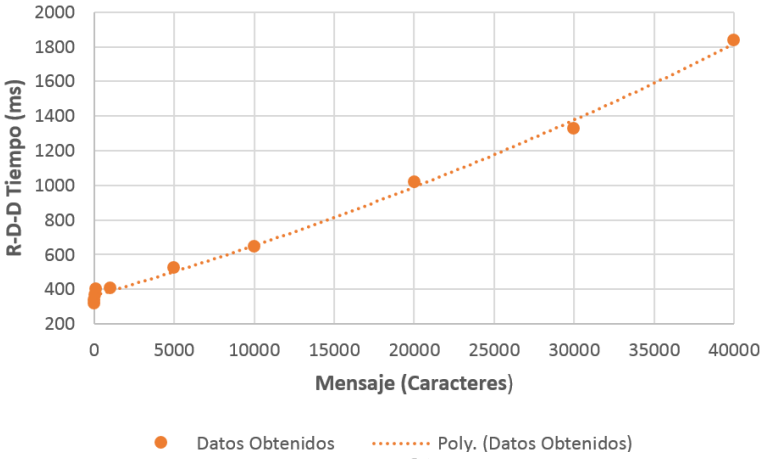
Tabla 1. Resultados del conjunto de pruebas.

MP3 Original	Mensaje [Caracteres]	Mensaje [Bytes]	Mensaje Comprimido [Bytes]	Mensaje Encriptado [Bytes]	Tamaño Objeto Cubierta [S]	ΔS	Duración Objeto Cubierta [T]	ΔT
1.mp3 T: 03:37 S: 6989701 [bytes]	10 100 1000 10000 20000 30000 40000	20 200 2000 20000 40000 60000 80000	31 134 843 7795 15514 23233 31697	32 144 848 7808 15520 23248 31712	7221908	232207	03:37	0
2.mp3 T: 03:06 S: 6217485 [bytes]	10 100 1000 10000 20000 30000 40000	20 200 2000 20000 40000 60000 80000	31 134 843 7795 15514 23233 31697	32 144 848 7808 15520 23248 31712	6217485	0	03:06	0
3.mp3 T: 04:13 S: 6189397 [bytes]	10 100 1000 10000 20000 30000 40000	20 200 2000 20000 40000 60000 80000	31 134 843 7795 15514 23233 31697	32 144 848 7808 15520 23248 31712	6356833	167436	04:13	0
4.mp3 T: 02:43 S: 2642491 [bytes]	10 100 1000 10000 20000 30000 40000	20 200 2000 20000 40000 60000 80000	31 134 843 7795 15514 23233 31697	32 144 848 7808 15520 23248 31712	2883724	241233	02:43	0
5.mp3 T: 03:00 S: 2916089 [bytes]	10 100 1000 10000 20000 30000 40000	20 200 2000 20000 40000 60000 80000	31 134 843 7795 15514 23233 31697	32 144 848 7808 15520 23248 31712	3165213	249124	03:00	0

La Figura 4b representa el tiempo que la aplicación requiere para la recuperación de la información embebida y cifrada, el tiempo de recuperación en función del número de caracteres muestra un comportamiento lineal. En términos de usabilidad, esta herramienta consta de dos interfaces gráficas de usuario, facilitando así la interacción entre el usuario y la aplicación. En cuanto a la seguridad, debemos evaluar la solidez del resultado cifrado, y la capacidad de embebido e imperceptibilidad del archivo MP3. La seguridad del texto cifrado reside en el algoritmo de encriptación y la contraseña. Como se ha mencionado antes, no hay cambio en el audio o la incorporación de ruidos o distorsiones. Por otro lado, la imagen utilizada para el proceso esteganográfico se mantiene visualmente intacta, por lo tanto, hay un buen nivel de imperceptibilidad.



(a)



(b)

Figura 4. Tiempos de procesamiento.

En general, la aplicación logra un funcionamiento satisfactorio, aunque existen algunas limitaciones que podrían mejorarse en trabajos futuros, entre éstos se encuentran el límite establecido en un número de caracteres (las técnicas de esteganografía que nos permiten incorporar más información tienen menos imperceptibilidad que otras donde menos información puede ocultarse). Estos resultados contrastan con los obtenidos en los trabajos presentados por Asad *et al.* (2012), Salman & Kanigoro (2014), Sing & Kumar (2016) e Indrayani, Nugroho, Hidayat & Pratama (2016), en los cuales no se indica el número máximo de bytes a embeber. Asad *et al.* (2012) y Salman & Kanigoro (2014), probaron sus aplicaciones con mensajes cortos, no se especifica el número máximo de caracteres. Sing & Kumar (2016) presenta breves resultados en los que afirman que el objeto de cubierta no se ve afectado por el tamaño o la duración. Los resultados obtenidos por Salman & Kanigoro (2014) muestran que el tamaño del objeto de cubierta es variable según la longitud del mensaje inicial. En cuanto al tiempo de procesamiento, la aplicación propuesta en este trabajo es más rápida, esto se debe a que la aplicación de Salman & Kanigoro (2014) está desarrollada para dispositivos móviles donde el procesador y la memoria son limitados y en ningún caso es comparable con una aplicación de escritorio. Por último, los métodos esteganográficos propuestos por Asad *et al.* (2012) e Indrayani *et al.* (2016) son bastante interesantes y podrían aplicarse al tag ID3v2 del archivo y no a las tramas de audio con resultados exitosos.

5. CONCLUSIONES

Este documento ha explicado en resumen el desarrollo de un software esteganográfico y de encriptación fácil de usar, capaz de encriptar y embeber texto en forma de un mensaje dentro de un archivo MP3, uno de los formatos de música más ampliamente utilizados alrededor del mundo. Además, la funcionalidad de descifrado y extracción se ha implementado dentro de la herramienta, contribuyendo al conjunto de aplicativos de software dedicadas a la protección de la confidencialidad de la información. El hecho de cifrar el mensaje antes del proceso esteganográfico disminuye la probabilidad de que el atacante obtenga el texto original. Si el adversario usa un método apropiado de esteganálisis o tiene la fortuna de detectar la presencia de información guardada dentro del objeto de cubierta, debe ser capaz de extraerlo completamente y descifrarlo usando el algoritmo criptográfico correcto y la contraseña. Si el mensaje incrustado no es extraído correctamente, o sufre una pequeña modificación, hará que el texto plano sea imposible de recuperar. El presente artículo puede tomarse como punto de partida para el desarrollo de herramientas mucho más poderosas con diversas técnicas de esteganografía trabajando juntas, aumentando la capacidad sin disminuir factores como la robustez, la imperceptibilidad, y además sin afectar las características intrínsecas del archivo de cubierta.

AGRADECIMIENTOS

Mi sincero agradecimiento a SENESCYT (Secretaria de Educación Superior Ciencia, Tecnología e Innovación) por hacer posible esta oportunidad de adquirir nuevos conocimientos y desarrollar nuevas habilidades en el extranjero durante el estudio de la Maestría.

REFERENCIAS

Anderson, R. J. (2008). *Security Engineering: A guide to building dependable distributed systems.* Wiley Publishing.

Asad, M., Gilani, J., & Khalid, A. (2012). *Three layered model for audio steganography*. In: 2012 International Conference on Emerging Technologies, pp. 1-6.

Atoum, M. (2015). A comparative study of combination with different LSB techniques in mp3 steganography. *Information Science and Applications*, pp. 551-560.

Bazyar, M., & Sudirman, R.(2014). A recent review of MP3 based steganography methods. *International Journal of Security and Its Applications, 8*(6), 405-414.

Daemen, J., & Rijmen, V. (2013). *The design of Rijndael: AES-the advanced encryption standard*. Springer Science & Business Media, 128 pp.

Delfs, H., & Knebl, H., (2001). *Introduction to cryptography: principles and applications*. New York: Springer-Verlag Inc., 367 pp.

Egidi, L., & Furini, M. (2005). Bringing multimedia contents into MP3 files. *IEEE Communications Magazine*, 43(5), 90-97.

Futcher, L., & von Solms, R. (2008). *Guidelines for secure software development*. In: Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology. ACM, pp. 56-65.

Indrayani, R., Nugroho, H., Hidayat, R., & Pratama, I. (2016). *Increasing the security of MP3 steganography using aes encryption and md5 hash function*. In: 2016 2nd International Conference on Science and Technology-Computer (ICST), pp. 129-132.

Maleki, N., Jalali, M., & Jahan, M. V. (2014). Adaptive and non-adaptive data hiding methods for grayscale images based on modulus function. *Egyptian Informatics Journal, 15*(2), 115-127.

Mathur, M., & Kesarwani, A. (2013). *Comparison between DES, 3DES, RC2, RC6, Blowfish and AES*. In: Proceedings of National Conference on New Horizons in IT-NCNHIT, Vol. 3, pp. 143-148.

Nilsson, M., Sundström, J. (2003). ID3v2.

Paar, C., & Pelzl, J., (2009). *Understanding cryptography: a textbook for students and practitioners*. Berlin: Springer Science & Business Media.

Petitcolas, F., Anderson, R., & Kuhn, M. (1999*). Information hiding - A survey.* In: Proceedings of the IEEE, 87(7), pp. 1062-1078.

Salman, A., & Kanigoro, B. (2014). Steganography application program using the ID3v2 in the MP3 audio file on mobile phone. *Journal of Computer Science, 10*(7), 1249-1252.