IMPLEMENTANDO ESTEGANOGRAFÍA EN IMÁGENES CIFRADAS

Implementando Esteganografía en Imágenes Cifradas

Ing. Beatriz Marlet Torres Perea

bmtorresp@gmail.com

CIDETEC-IPN

Abstract

Este artículo está enfocado en describir e implementar la técnica de esteganografía basada en LSB para ocultar una imagen cifrada dentro de otra imagen no cifrada, sin pérdida de información. Se usó una imagen cifrada previamente ya que el objetivo es describir la técnica LSB en imágenes. Para la implementación se utilizó el lenguaje de programación Java.

Introducción

A lo largo de la historia los seres humanos han visto la necesidad de ocultar información confidencial con el único objetivo de mantener a salvo aquellos de su interés, por ejemplo: políticos, sociales, personales, económicos, etc.

En los últimos años los ataques cibernéticos han aumentado en forma gradual, tan solo en el 2015 México sufrió más de 400 ataques cibernéticos, colocándose en la posición 24 a nivel mundial [1], mientras que en 2017 a nivel mundial se vivió uno de los peores ataques de la historia [2] con ello resguardar dicha información de ataques se convierte en todo un reto.

El propósito de la esteganografía es ocultar información y que dicha información no sea detectada a simple vista, una de las técnicas más populares es la de sustitución LSB [3]. En este sentido la información a ocultar será una imagen previamente cifrada, lo cual permitirá incrementar la complejidad del sistema ya que si ocultamos solo la imagen es posible realizar búsquedas específicas para hallar dicha imagen y reconstruirla, sin embargo, si ocultamos y además ciframos la imagen, aunque se pueda reconstruir la imagen cifrada si se desconoce el algoritmo de cifrado no será posible reconstruir la imagen.

Esteganografía

Esteganografía proviene del griego “steganos” que significa cubierto u oculto y “graphein” que significa escritura [4]. Por lo tanto, es posible definir esteganografía como la ciencia que estudia los procedimientos encaminados a ocultar la existencia de un mensaje en lugar de ocultar su contenido. [5]

Desde el siglo V, se tiene evidencia que el griego Heródoto de Halicarnaso implementaba técnicas para enviar mensajes de manera encubierta [6]. Con el auge de las Tics las técnicas de esteganografía también evolucionaron.

Los estegosistemas modernos se clasifican principalmente en dos categorías:

- Estegosistemas de clave simétrica:
  Son aquellos los cuales tanto el emisor como el receptor comparten la misma llave, por lo tanto, la seguridad de este sistema prevalece en dicha llave. [7]
- Estegosistemas de clave pública:
  Son aquellos los cuales requieran tanto la llave que comparte el emisor y el receptor que es conocida como llave pública y otra llave la cual se conoce como llave privada la cual tanto el emisor como el receptor tienen una de manera independiente [7].

Descripción de la técnica de sustitución LSB

La técnica de sustitución LSB por sus iniciales en inglés significa: Least Significant Bit (bit menos significativo), es el mecanismo de ocultación clásico en imágenes digitales, con un grado de seguridad adecuado debido a que LSB permite ocultar gran cantidad de información, como mínimo 1 bit por cada pixel de la imagen. Otro motivo para usar LSB es que la implementación del algoritmo es sencilla [7].

Antes de describir la técnica LSB es importante considerar los siguientes puntos que se deducen de la composición de una imagen:

- Un byte tiene ocho bits.
- Un pixel se compone de RGB (Red, Green, Blue) cada componente de color se compone de un byte.
- Un pixel tiene tres bytes.
- Si una imagen tiene 512 pixeles de ancho y 512 pixeles de alto, significa que se tienen 512*512*3= 786 432 bytes.
- Para ocultar un componente de color se requieren 8 bytes.
- Para guardar un pixel completo se requieren 24 bytes ó bien ocho pixeles.

La técnica LSB realiza una sustitución secuencial de los bits en la codificación de los pixeles de una imagen, ya sea de una posición fija de la imagen o se puede calcular [7]. Una vez conocidos los pixeles disponibles y su codificación se extrae el bit menos significativo de cada byte que compone cada pixel, y se inserta el nuevo valor a ocultar en el bit menos significativo.

En la tabla 1, se describe con un ejemplo dicha sustitución de bits. Se pretende ocultar el valor en decimal 27 que representado en binario es 00011011. En esta tabla se encuentra separado cada pixel con sus tres componentes RGB con sus valores originales tanto en su representación en decimal como en binario, en las últimas dos filas de la tabla se ha aplicado la técnica LSB sustituyendo el bit menos significativo de cada byte para ocultar el valor 27 (00011011). Se observa también que la componente B, del pixel 3 no fue usado ya que para ocultar el valor 27 solo se requieren 8 bytes. En este ejemplo solo cambiaron 3 bytes de los 8 que se usaron.

Tabla 1. Ejemplo de la técnica LSB.

Pixel	Componente de color	Valor en decimal	Valor en binario	Valor en binario aplicando LSB	Valor en decimal aplicando LSB
Pixel 1	R	32	0010000**0**	0010000**0**	32
	G	22	0001011**0**	0001011**0**	22
	B	12	0000110**0**	0000110**0**	12
Pixel 2	R	88	0101100**0**	0101100**1**	89
	G	51	0011001**1**	0011001**1**	51
	B	21	0001010**1**	0001010**0**	20
Pixel 3	R	25	0001100**1**	0001100**1**	25
	G	30	0001111**0**	0001111**1**	31
	B	23	0001011**1**	0001011**1**	23

Implementación de la técnica LSB

El diseño propuesto del sistema se representa en el diagrama 1, el cual se compone de la imagen a cifrar, el cual al aplicarse el algoritmo de cifrado se obtiene una imagen cifrada, dicha imagen cifrada es la que se oculta en la imagen que le llamaremos imagen base, finalmente el resultado es la imagen con esteganografía.

Diagrama 1. Diseño propuesto del sistema.

Para el cifrado de la imagen se puede proponer algún algoritmo de cifrado simétrico como DES, Triple DES, AES, o cualquier otro algoritmo con algún hibrido usando RSA, El Gamal o La Curva Elíptica.

En este artículo no se abordará la técnica de cifrado solo se entenderá que se implementa algún algoritmo de los antes mencionados, ya que él estudió de la criptografía no es el tema principal del artículo.

La imagen a cifrar se puede observar en la figura 1, en la figura 2 se observa la imagen cifrada. Ambas imágenes tienen las siguientes características: 512 pixeles de ancho y 512 pixeles de alto.

Figura 1. Imagen a cifrar.

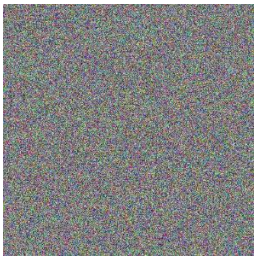

Figura 2. Imagen Cifrada.

La implementación del diseño propuesto se llevó a cabo con el IDE de Netbeans y Java como lenguaje de programación los pasos que se usaron son los siguientes:

1. Lectura de la imagen a cifrar.
2. Implementación de la técnica de cifrado.
3. Lectura de la imagen cifrada, se guarda cada componente de color en un conjunto de bytes.
4. Lectura de la imagen base, se guarda cada componente de color en un conjunto de bytes.
5. Comparar si la imagen base es lo suficientemente grande para que la imagen cifrada pueda ser oculta.
6. Leer el primer byte a ocultar.
7. Leer el primer byte de la imagen base.
8. Remplazar el bit menos significativo en el byte de la imagen base (se coloca el primer bit de la imagen a ocultar empezando por el bit más significativo).
9. Continuar con el siguiente byte de la imagen base y repetir el paso 6 ocho veces, ya que es el tamaño del byte.
10. Leer el siguiente byte a ocultar, así como el siguiente byte de la imagen base, repetir el paso 6 y 7 hasta terminar con los bytes de la imagen a ocultar.
11. Generar la nueva imagen.

En la figura 3, se muestra la imagen base antes de implementar el algoritmo LSB, mientras que la figura 4, es el resultado de la imagen después de usar LSB.

Figura 3. Imagen base.

Figura 4. Imagen con LSB.

Para obtener la imagen cifrada, se realiza la lectura de la imagen con LSB y en cada byte se obtiene el bit menos significativo, hasta formar un byte. Lo anterior se repite hasta recuperar los bytes necesarios y formar la imagen.

Para formar la imagen se requiere saber el ancho y alto de la imagen para lo cual en los primeros 16 bytes de la imagen con LSB se reservan para guardar dicha información.

Conclusiones

La técnica LSB, tiene diversos ataques ya que es posible reconstruir una imagen quitando el bit menos significativo de cada byte de información, sin embargo, aunque se pueda obtener la imagen cifrada si no se conoce el tipo de algoritmo con el cual se cifró la imagen será imposible reconstruirla.

Es posible darle fortaleza al algoritmo LSB realizando alguna modificación para guardar el bit de manera aleatoria en toda la imagen y no de manera consecutiva como se implementa hoy en día.

La arquitectura del sistema permite que la imagen pueda ser enviada por cualquier medio de información.

Referencias

[1] Anónimo (2016). México sufre más de 400 ataques cibernéticos durante el 2015. De El Universal Sitio web: http://www.eluniversal.com.mx/articulo/techbit/2016/03/4/mexico-sufre-mas-de-400-ataques-ciberneticos-durante-el-2015.

[2] Anónimo. (2017). Las lecciones que nos dejó el ataque de 'WannaCry'. Marzo 28,2018, de Excelsior Sitio web: http://www.excelsior.com.mx/hacker/2017/06/14/1169706.

[3]Areitio J. (2008). Esteganografía y Estegoanálisis. En Seguridad de la información. Redes, informática y sistemas de información (406). España: Editorial Paraninfo.

[4] Ribagorda, A. (2004). Avances en criptología y seguridad de la información. España: Diaz de Santos.

[5] (2003). Esteganografía. 12-02-2016, de Ibiblio Sitio web: http://ibiblio.org/pub/Linux/docs/LuCaS/Manuales-LuCAS/doc-unixsec/unixsec-html/node320.html.

[6] Núñez, L. (1994). Manual de paleografía. España: Catedra.

[7] Muñoz, A. (2017). Ocultación de información en imágenes digitales. En Privacidad y ocultación de información digital Esteganografía Protegiendo y atacando redes informáticas (p.38, pp.67-68, p.81). Madrid, España: RA-MA.