Modelado y Programación

Esteganografía por el método LSB

Introducción

La esteganografía es el conjunto de técnicas algorítmicas para ocultar un conjunto de datos en otro de forma que los primeros no sean evidentes, permanezcan íntegros y sean recuperables mientras que los segundos, a los que se les denomina *portadores*, no deben verse alterados de manera evidente. Un método común es ocultar mensajes de texto claro en archivos de imágenes. Se aprovecha en este caso el hecho de que, con frecuencia, las imágenes no requieren de preservar todos los datos que contienen para ser útiles.

Un método usado para almacenar un archivo de texto en una imagen consiste en usar el bit menos significativo de cada byte de datos de la imagen para guardar un bit de los datos a ocultar. Por supuesto se supone que la cantidad de bytes de datos de la imagen es suficiente para guardar el archivo de texto respectivo. Típicamente cada pixel de una imagen es una terna (o cuarteta si existe un canal alfa para la transparencia) de bytes, uno para especificar el tono del pixel en el canal rojo, otro para el canal verde y un tercero para el canal azul. Si se cambia el valor del bit menos significativo de cada uno de estos canales, el color del pixel será cambiado de manera imperceptible para quien observa a simple vista la imagen, así que por cada pixel hay, el menos, tres bits útiles para almacenar datos ocultos.

Por supuesto se está suponiendo que, una vez puestos los bits de datos a ocultar en la imagen, estos permanecerán en la imagen de manera íntegra, por lo que no se deberán usar formatos de compresión con pérdida para guardar las imágenes que contienen datos ocultos usando este esquema.

Entrada

El programa debe funcionar en dos modalidades, para ocultar (opción **h**) y para develar (opción **u**).

Ocultar. Se debe proporcionar, en la línea de llamada:

1. La opción **h**.
2. El nombre del archivo que contiene el texto a ocultar.
3. El nombre del archivo de imagen.
4. El nombre del archivo de imagen resultante con los datos ocultos.

Develar. Se debe proporcionar, en la línea de llamada:

1. La opción **u**.
2. El nombre del archivo con la imagen que contiene los datos ocultos.
3. El nombre del archivo en el que se guardará el texto develado.

Salida

Dependiendo de la opción seleccionada el programa debe entregar lo que se especifica.

Ocultar.

El archivo de imagen con el texto oculto.

Develar.

El archivo con el texto claro.

Proceso

Un formato ampliamente usado en Internet y que almacena las imágenes sin pérdida de datos es PNG, al que, adicionalmente, se le puede agregar un canal alfa de transparencia, con lo que cada pixel puede almacenar 4 bits (un nibble) de datos, es decir medio byte. Si además se reduce el alfabeto del texto de entrada a un byte por cada caracter, entonces dos pixeles bastan para almacenar un caracter completo. Esta restricción es válida si lo que se quiere ocultar es un texto que debe ser entendible por seres humanos. Si prescindimos de algunos signos y de los caracteres con tilde del español, podemos usar los códigos desde el 32 hasta el 127 de la tabla de códigos (en ese rango coinciden la antigua ASCII, la de UTF-8 y, claro, la de Unicode). Es posible restringir aún más el alfabeto si no se incluyen letras minúsculas, por ejemplo. El programador puede decidir usar el mínimo número de bits y de esta manera aprovechar mejor el espacio disponible en la imagen.[1]

Implementación

Para la implementación se pueden utilizar bibliotecas de software libre disponibles para manipular imágenes en diversos formatos. Las innovaciones que redunden en mejor fortaleza esteganográfica serán tomadas en cuenta.

[1] Y no sólo eso. Si se usa la tabla de códigos tal como se describe, por ejemplo, el primer bit siempre será cero, por lo que el canal rojo de toda la imagen siempre tendrá números pares, lo que podría resultar sospechoso para alguien que analice la imagen.