

Esteganografía: ocultamiento de información en entornos digitales

Seguridad en Sistemas
Licenciatura en Ciencias de la Computación

Prof. Lic. Leonardo de Matteis
Estudiante Ludmila Prolygin

Julio de 2025

Tabla de contenidos

Objetivos	3
Introducción	3
Historia	3
Diferencia con criptografía	4
Relevancia en la seguridad	4
Métodos y herramientas	4
Técnicas generales de ocultamiento y estegoanálisis	4
Esteganografía en imágenes	5
Esteganografía en audio y video	6
Esteganografía en protocolos de comunicación UDP	6
Esteganografía en sistema de archivos	6
Esteganografía en texto	7
Implementación práctica	7
Preparación del entorno	8
Esquematización del código	9
Invocación del programa	9
Resultados	9
Ocultamiento de texto	10
Ocultamiento de un archivo PDF	10
Ocultamiento de un archivo de video	11
Análisis de los resultados	11
Comparación con herramientas	12
Steghide	13
Strings	13
Binwalk	13
Casos de uso y aplicaciones actuales	13
Comunicación segura	13
Protección de derechos de autor mediante esteganografía y marca de agua oculta	14
Protección de datos médicos en imágenes clínicas	14
Espionaje gubernamental	14
Comunicaciones terroristas encubiertas	14
Distribución de malware	15
Stegosploit	15
Consideraciones ético-legales	16
Conclusión	16
Repositorio	17
Referencias y recursos utilizados	17

Objetivos

El proyecto se centrará en el estudio de la esteganografía como técnica de ocultamiento de información, diferenciándose de otras estrategias de protección. Se analizarán los fundamentos teóricos, así como las principales formas en que puede aplicarse en la práctica, tanto en contextos legítimos como en situaciones de uso malicioso.

El trabajo incluirá una reflexión sobre el rol que ocupa la esteganografía en el campo de la seguridad, considerando su potencial como herramienta de protección de datos, pero también los riesgos asociados.

A modo de cierre, se propondrá una exploración práctica que permita demostrar el funcionamiento de este tipo de técnicas en un entorno controlado, con el objetivo de vincular los conceptos teóricos con su aplicación concreta.

Se busca así no solo comprender los aspectos técnicos y conceptuales de la esteganografía, sino también generar una mirada crítica sobre su uso en distintos contextos, destacando su relevancia dentro de las problemáticas actuales en seguridad.

Introducción

La esteganografía es la ciencia que estudia y desarrolla métodos para ocultar información dentro de un medio portador —como una imagen, un archivo de audio o video— de manera que el mensaje permanezca invisible para los receptores no intencionados. A diferencia de la criptografía, que oculta el contenido de un mensaje, la esteganografía oculta su existencia misma.

Para lograr este ocultamiento, la esteganografía aprovecha la redundancia presente en los archivos portadores, permitiendo injectar información sin generar alteraciones perceptibles a simple vista o al oído.

Historia

El término esteganografía deriva del griego *stegano* que significa "oculto" y *graph* que significa "escrito".

Es una técnica utilizada desde hace siglos. En la antigua Grecia las personas solían tallar mensajes en madera y cubrirla con cera para ocultar el mensaje. Los romanos, por su parte, utilizaban tintas invisibles que, al ser expuestas al calor, revelaban el mensaje. Durante las guerras mundiales del siglo XX, la esteganografía también tuvo un papel destacado, siendo usada para ocultar mensajes en microfilm, fotografías o incluso en tejidos de ropa.

Si bien estas aplicaciones distan de la esteganografía digital conocida al día de hoy, se fundan en el mismo concepto: ocultar la existencia de información sensible a plena vista, haciendo que pase inadvertida para quienes no son sus destinatarios.

Diferencia con criptografía

La criptografía es el campo que se ocupa del diseño, análisis y utilización de algoritmos para el cifrado y descifrado de información; *arte de escribir con clave secreta o de un modo enigmático*.

Su objetivo principal es proteger el contenido de un mensaje, de modo que, aunque sea interceptado por un tercero, no pueda ser comprendido sin conocer la clave correspondiente; considera la posibilidad de la intercepción de los datos, pero la seguridad está provista por una clave de descifrado necesaria para desencriptar el mensaje.

Mientras que la criptografía hace evidente que hay un mensaje cifrado, la esteganografía pretende pasar completamente desapercibida. Por eso, ambas técnicas pueden considerarse complementarias: es posible cifrar un mensaje y luego ocultarlo mediante esteganografía, añadiendo una capa extra de seguridad.

Relevancia en la seguridad

La esteganografía desempeña un papel fundamental en el ámbito de la seguridad informática, especialmente en contextos donde mantener la privacidad y confidencialidad de las comunicaciones resulta crucial. Al esconder la existencia misma del mensaje, ofrece una protección adicional a la proporcionada por métodos criptográficos convencionales. Esto la convierte en una técnica especialmente valiosa en situaciones de alta sensibilidad. Sin embargo, su relevancia también ha crecido por su potencial uso malicioso: la posibilidad de ocultar y ejecutar código malicioso encubierto dentro de archivos aparentemente inofensivos representa una amenaza real que desafía los métodos tradicionales de detección y análisis.

Métodos y herramientas

Para llevar a cabo el ocultamiento de información mediante esteganografía, es necesario tener en cuenta tres aspectos fundamentales: el medio portador que será utilizado, el contenido y formato del mensaje a ocultar, y la técnica específica empleada.

Como fue mencionado anteriormente, algunos de los medios más comunes incluyen imágenes (PNG, BMP), audio (MP3, WAV), video (MP4, AVI) y archivos de texto. El tipo de medio utilizado determinará, en gran medida, la técnica de ocultamiento más adecuada.

Otro aspecto esencial es la información a ocultar. Esta puede ser texto simple, documentos, archivos cifrados o no cifrados, dependiendo del nivel de seguridad requerido.

Técnicas generales de ocultamiento y estegoanálisis

Así como existen herramientas destinadas al ocultamiento de información, también se han desarrollado diversas soluciones enfocadas en la detección y análisis forense de contenido esteganográfico, disciplina conocida como estegoanálisis.

- Least Significant Bit (LSB): Consiste en modificar los bits menos significativos de cada byte del medio portador para insertar información secreta. Esta técnica es frecuentemente usada en imágenes y audio.
- Dominio de frecuencia: En lugar de modificar directamente los datos crudos (como píxeles o muestras de audio), la información secreta se incorpora en los componentes de frecuencia del archivo mediante transformaciones como la Transformada Coseno Discreta (DCT) o la Transformada Wavelet Discreta (DWT).
- Planos de bits (Bit Planes): Los datos secretos se ocultan en planos de bits de orden superior en las imágenes, resultando potencialmente más seguro debido a que se manipulan bits menos perceptibles.

Esteganografía en imágenes

La esteganografía en imágenes implica técnicas especializadas debido a la naturaleza visual del medio portador.

- Sustitución del bit menos significativo (LSB). Se modifica el bit menos significativo de los píxeles. Por ejemplo, en una imagen de color de 8 bits, al alterar el bit menos significativo de cada componente (Rojo, Verde, Azul), el cambio visual es casi imperceptible.
- Enmascarado y filtrado. Se aplican máscaras o filtros sobre partes importantes de la imagen, insertando datos de manera que los cambios sean difíciles de detectar visualmente. Generalmente se combina con otras técnicas.
- Técnicas de dominio transformado. La información se incorpora en la imagen transformada al dominio de frecuencias. Esto incrementa la resistencia frente a la compresión y el ruido. Es una forma más compleja de ocultar mensajes en una imagen debido a que usa diferentes algoritmos y transformaciones sobre esa imagen. Los datos son incorporados en los coeficientes de cambio, transformados dentro del área de frecuencia mediante diversos métodos (por ejemplo, mediante transformadas como DCT, DFT o DWT).
- Técnicas basadas en paletas. Se manipula la paleta de colores en imágenes con indexación de color, típicamente en formatos como GIF, alterando levemente las entradas del índice de color para codificar el mensaje oculto.
- Sustitución de segmentos de imagen. Consiste en reemplazar segmentos completos de una imagen con otros que contienen datos ocultos, cuidando que los nuevos segmentos sean visualmente similares al original.
- Codificación de píxeles redundantes. Se aprovecha la redundancia visual en las imágenes para insertar información secreta, manteniendo los cambios lo más mínimos y discretos posibles.
- Pixel Value Differencing (PVD). Esta técnica selecciona dos píxeles consecutivos en una imagen para esconder los datos. La información a ocultar está determinada al comprobar la diferencia entre ambos píxeles.

La elección del formato del archivo portador afecta significativamente la técnica de ocultamiento, debido a que diferentes formatos poseen características propias. Por ejemplo, las imágenes en formato JPEG utilizan un método de compresión con pérdida (lossy compression), lo que implica que ciertos detalles visuales del archivo original son eliminados para reducir el tamaño. Esto limita la capacidad de ocultar información usando técnicas

como el LSB, ya que los datos ocultos podrían perderse o distorsionarse durante la compresión. En contraste, los formatos sin pérdida (lossless), como PNG o BMP, conservan intactos todos los detalles del archivo original, ofreciendo mayor fiabilidad y estabilidad al ocultar mensajes mediante la modificación directa de bits o píxeles, facilitando así técnicas como la sustitución LSB o la codificación de píxeles redundantes.

Esteganografía en audio y video

Los archivos de sonido también permiten ocultar información. A diferencia de los archivos de imagen, el ser humano sólo escucha un rango de frecuencia. Puede utilizarse, por ejemplo, la técnica LSB anteriormente mencionada.

Los archivos de video generalmente son una colección de imágenes y sonidos, por lo que la mayoría de técnicas presentadas en imágenes y videos y audio se puede aplicar a archivos de video. Usualmente se utiliza el método DCT que cambia ligeramente cada una de las imágenes del video hasta el punto que el ojo humano no lo perciba. La ventaja de aplicar esteganografía en video es la gran cantidad de datos que puede ocultarse al interior por el hecho del flujo continuo de imágenes y sonido en movimiento. Cualquier distorsión pequeña puede pasar aparentemente desapercibida por el hombre.

Esteganografía en protocolos de comunicación UDP

El protocolo UDP (User Datagram Protocol) opera en la capa de transporte y se caracteriza por ser no orientado a conexión; esto significa que no requiere que se establezca previamente una conexión entre emisor y receptor para transmitir datos. A diferencia de otros protocolos, UDP no ofrece garantías sobre la entrega de información: los datos podrían perderse, duplicarse o llegar fuera del orden original, siendo por tanto considerado un protocolo no fiable.

Debido a sus características, el campo del puerto de origen dentro del paquete UDP puede ser aprovechado para manipular información, ya que este campo permite indicar al receptor por qué puerto el emisor está enviando los datos. De este modo, resulta posible ocultar hasta 16 bits por cada paquete UDP enviado. Esta técnica es especialmente útil en escenarios donde el receptor no necesita conocer el puerto origen de la comunicación, como sucede habitualmente en transmisiones unidireccionales del emisor hacia el receptor.

Esteganografía en sistema de archivos

Desde una perspectiva física, los archivos son almacenados en bloques dentro del sistema de archivos. Dado que el tamaño de un archivo no siempre coincide exactamente con el tamaño de estos bloques, suele quedar un espacio sin utilizar dentro del bloque asignado al archivo, denominado fragmentación interna. Este espacio puede aprovecharse para ocultar información de forma efectiva, ya que resulta invisible tanto para el sistema operativo como para las herramientas comunes de validación o análisis de archivos.

Entre sus ventajas destaca precisamente su invisibilidad frente a sistemas convencionales. Sin embargo, tiene como principal desventaja la portabilidad, ya que la información oculta en el espacio de fragmentación interna se pierde fácilmente al realizar copias con

herramientas estándar o si el archivo cambia de tamaño. Por ello, es recomendable utilizar esta técnica con archivos cuya longitud se sepa que no variará, como pueden ser los archivos propios del sistema operativo.

Esteganografía en texto

La esteganografía en texto se basa en ocultar información dentro de documentos escritos, aprovechando las propiedades visuales, tipográficas o semánticas del texto. Dado que el texto ofrece poca redundancia comparado con otros medios (como imágenes o audio), las técnicas utilizadas deben ser especialmente sutiles para evitar ser detectadas.

- Inserción de caracteres invisibles. Se utilizan caracteres Unicode invisibles, como espacios de ancho cero (U+200B) o caracteres especiales de formato, para codificar bits de información. Como estos caracteres no afectan la apariencia del texto, el contenido visual no se altera, aunque al copiar o analizar el archivo pueden detectarse fácilmente si se utilizan herramientas adecuadas.
- Codificación mediante espacios o tabulaciones. Algunos métodos asignan significado a la cantidad o tipo de espacio entre palabras, líneas o párrafos. Por ejemplo, un espacio simple puede representar un 0 y un espacio doble un 1. Esta técnica es sencilla pero muy frágil, ya que se pierde fácilmente al formatear el texto o copiarlo a otro formato.
- Manipulación de formato. Se oculta información variando discretamente el tamaño de fuente, el estilo (negrita, cursiva), el color de la letra o el interlineado. Estas variaciones, aunque perceptibles al ojo humano, pueden pasar desapercibidas si se aplican en fragmentos pequeños y bien distribuidos.
- Codificación en errores ortográficos o gramaticales intencionados. Se introduce un patrón de errores sutiles (como omitir tildes o cambiar letras) para representar información binaria, basándose en reglas definidas previamente.

Este tipo de esteganografía es especialmente vulnerable a ediciones, correcciones automáticas, conversión de formato o cambios de codificación, por lo que no es recomendable cuando se desea una alta robustez o persistencia del mensaje oculto.

Implementación práctica

Con el objetivo de comprender mejor el funcionamiento y alcance de la esteganografía digital, se realizó una implementación práctica mediante la creación de un script en Python. El objetivo del mismo fue ocultar tanto mensajes de texto como archivos completos dentro de imágenes, utilizando la técnica conocida como sustitución del bit menos significativo (LSB).

- Ocultamiento de mensajes de texto. Se lee un mensaje ingresado por el usuario, se transforma en formato binario y posteriormente se insertan los bits del mensaje en los bits menos significativos de cada píxel de una imagen portadora (generalmente PNG por su estabilidad frente a modificaciones en píxeles).
- Ocultamiento de archivos completos. Además del texto, el script soporta la ocultación de archivos enteros. El archivo seleccionado por el usuario es convertido

a formato binario y luego insertado en la imagen, siguiendo un proceso similar al usado para texto.

- Extracción del contenido oculto. También se implementó la funcionalidad de extracción, que permite recuperar el mensaje oculto o archivo desde la imagen portadora, verificando así la integridad y confidencialidad del proceso.

Durante las pruebas realizadas, se comprobó que la información oculta resultó imperceptible visualmente, cumpliendo así con uno de los principios fundamentales de la esteganografía. Se observó también que el proceso de extracción funcionó correctamente en todos los casos, siempre y cuando la imagen no sufriera modificaciones o compresiones con pérdida después de la inserción de los datos.

Preparación del entorno

Para llevar a cabo la implementación práctica del script de esteganografía, fue necesario configurar previamente un entorno adecuado.

Se utilizó Visual Studio Code como entorno de desarrollo para el código, y Python (versión 3.13.5 con pip versión 25.1.1) como lenguaje de programación.

Instalacion de librerias:

```
PS C:\Users\mlpro\Documentos\Universidad\seguridad-en-sistemas\Proyecto> pip3 install opencv-python numpy
Collecting opencv-python
  Downloading opencv_python-4.11.0.86-cp37-abi3-win_amd64.whl.metadata (20 kB)
Collecting numpy
  Downloading numpy-2.3.1-cp313-cp313-win_amd64.whl.metadata (60 kB)
  Downloading opencv_python-4.11.0.86-cp37-abi3-win_amd64.whl (39.5 MB)
    39.5/39.5 MB 1.1 MB/s eta 0:00:00
  Downloading numpy-2.3.1-cp313-cp313-win_amd64.whl (12.7 MB)
    12.7/12.7 MB 1.1 MB/s eta 0:00:00
Installing collected packages: numpy, opencv-python
Successfully installed numpy-2.3.1 opencv-python-4.11.0.86
```

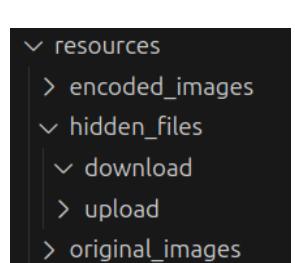
En caso de error:

```
ludmila-prolygin@ludmila-prolygin-A520M-ITX-ac:~$ sudo pip3 install opencv-python numpy
error: externally-managed-environment

  x This environment is externally managed
  ↳ To install Python packages system-wide, try apt install
     python3-xyz, where xyz is the package you are trying to
     install.
```

ejecutar el comando sudo apt install python3-opencv python3-numpy

Se optó por una jerarquía de directorios como la ilustrada a continuación.



- resources contiene las carpetas que contienen los diferentes archivos que interactúan con el código.
 - encoded_images contiene la imagen resultado luego de incrustar información.
 - hidden_files contiene dos carpetas: upload contiene aquellos archivos que desean ser incrustados y, download, los archivos resultantes al realizar el decoding de la imagen.

Esquematización del código

```
def to_bin(data):
```

Esta función codifica la entrada `data` en su codificación binaria de 8 bits (1 byte), según el tipo de dato. La representación a utilizar es la de 8 bits puesto que la codificación RGB utiliza 8 bits por canal (3 bytes en total por pixel, un byte por color).

```
def encode(image_name, secret_data, n_bits=2):
```

Esta función incrusta `secret_data` dentro de la imagen utilizando la técnica LSB. Realiza esto considerando `n_bits` que representan la cantidad de bits que pueden modificarse por byte; el valor puede fluctuar de 1 a 6 según el tamaño del archivo, considerando que, a mayor tamaño, más distorsión presentará la imagen.

```
def decode(image_name, n_bits=1, in_bytes=False):
```

Esta función permite extraer el mensaje oculto en una imagen. `in_bytes` permite determinar si debe reconstruirse un mensaje como cadena de caracteres o como archivo.

```
def set_args():
```

Se determinan los argumentos posibles a utilizar al invocar el programa.

- `-t, --text`. El texto que quiere ocultarse en la imagen, solo debería especificarse para el encoding.
- `-f, --file`. El archivo que quiere ocultarse en la imagen, solo debería especificarse para el encoding.
- `-e, --encode`. Encoding de la imagen.
- `-d, --decode`. Decoding de la imagen.
- `-b, --bits`. Cantidad de LSB al hacer el encoding.

Invocación del programa

Por línea de comandos se debe invocar el programa con los argumentos correspondientes.

- Encoding.
 - Archivo. `python3 <programa> -e <imagen_origen> -f <archivo_a_ocultar> -b <cant_bits_a_modificar>`
 - Texto. `python3 <programa> -e <imagen_origen> -t <texto_a_ocultar> -b <cant_bits_a_modificar>`
- Decoding.
 - Archivo (se realiza una descarga). `python3 <programa> -d <imagen_encoded> -f <archivo_decoded> -b 1 <cant_bits_modificados>`
 - Texto (se imprime por línea de comandos). `python3 <programa> -d <imagen_encoded> -b 1 <cant_bits_modificados>`

Resultados

Durante la etapa de pruebas se evaluó el funcionamiento del script desarrollado para ocultar y recuperar información utilizando la técnica de sustitución de bits menos significativos (LSB) sobre distintos tipos de archivos: texto plano, un documento PDF y un archivo de video. En todos los casos se emplearon imágenes en formato PNG como archivos portadores.



Ocultamiento de texto

Se codificó un mensaje de texto corto utilizando 1 bit LSB por componente de color. El resultado fue exitoso: la imagen resultante no mostró diferencias perceptibles respecto de la original y la extracción posterior del mensaje fue precisa y sin pérdida de datos. Este caso representa un uso típico de esteganografía para comunicación sencilla y efectiva.



Imagen resultante al ocultar texto

Ocultamiento de un archivo PDF

Se ocultó un archivo PDF (de tamaño moderado) usando también 1 bit LSB. Si bien el tiempo de procesamiento fue ligeramente mayor que con el texto plano, el procedimiento fue exitoso. La recuperación del archivo fue completa y el contenido íntegro del PDF se mantuvo sin alteraciones. El archivo portador resultante no presentó distorsiones visibles.



Imagen resultante al ocultar un archivo PDF

Ocultamiento de un archivo de video

En esta prueba se ocultó un pequeño archivo de video utilizando 7 bits LSB, lo que permitió almacenar una mayor cantidad de datos. Si bien la imagen portadora fue modificada visualmente de manera notable (se observaron patrones de ruido), el video pudo recuperarse correctamente. Este resultado evidencia la relación directa entre la profundidad de bits utilizada y el impacto visual, además de mostrar los límites de capacidad de este tipo de técnica.



Imagen resultante al ocultar un archivo de video

Análisis de los resultados

Los resultados obtenidos permiten reflexionar sobre el equilibrio entre capacidad de ocultamiento y perceptibilidad visual. En los casos donde se utilizó 1 bit LSB, tanto para ocultar texto como un archivo PDF, las imágenes resultantes no presentaron cambios perceptibles a simple vista. Esta baja alteración se debe a que el bit menos significativo de cada componente de color (rojo, verde y azul) representa una variación mínima en el valor total del píxel, generalmente imperceptible para el ojo humano. En las siguientes imágenes

se muestra la variación de colores con la modificación de un LSB en los canales G (green) y B (blue).

	<input type="checkbox"/> <code>rgb(202, 146, 127)</code>
	<input type="checkbox"/> <code>#ca927f</code>
	<input type="checkbox"/> <code>rgb(202, 147, 126)</code>
	<input type="checkbox"/> <code>#ca937e</code>

Sin embargo, al aumentar la cantidad de bits modificados —como en la prueba con el archivo de video usando 7 bits LSB— los cambios se volvieron evidentes. Las alteraciones afectaron directamente la composición de color de los píxeles, generando patrones de ruido o distorsión que resultan fácilmente detectables visualmente. Esto pone en evidencia una de las principales limitaciones de las técnicas de esteganografía LSB: cuanto mayor es la capacidad de datos ocultos, menor es la imperceptibilidad. En las siguientes imágenes se muestra la variación de colores con la modificación de 7 (siete) LSBs en los canales G (green) y B (blue).

	<input type="checkbox"/> <code>rgb(202, 146, 127)</code>
	<input type="checkbox"/> <code>#ca927f</code>
	<input type="checkbox"/> <code>rgb(202, 237, 0)</code>
	<input type="checkbox"/> <code>#caed00</code>

Desde el punto de vista perceptual, el ojo humano es más sensible a cambios en ciertas gamas (como el verde) y menos en otras (como el azul), lo cual podría aprovecharse para mejorar el ocultamiento selectivo. Asimismo, el uso de imágenes con muchas áreas de color plano o baja variación facilita la detección de alteraciones, mientras que imágenes con alto detalle o textura natural "disfrasan" mejor la manipulación.

Estas observaciones sugieren que para lograr un ocultamiento eficaz y discreto, es fundamental elegir no solo el número de bits a modificar, sino también una imagen portadora adecuada. Además, sería interesante considerar métodos adaptativos que analicen el contenido visual y distribuyan la información oculta en regiones menos sensibles al cambio.

Comparación con herramientas

Existen diversas herramientas disponibles para realizar esteganografía, tanto de código abierto como privativas. Algunas permiten ocultar información en imágenes, audio o video, utilizando técnicas similares a las implementadas en este proyecto y otras ofrecen métodos más sofisticados.

Si bien existen infinidad de herramientas, a lo largo del desarrollo del presente proyecto se utilizaron 3 (tres): Steghide, Strings y Binwalk. Todas ellas fueron utilizadas en un entorno virtual con Kali Linux como sistema operativo base.

Durante las pruebas realizadas, se observó que todas las herramientas analizadas fueron más complejas de utilizar en comparación con el script desarrollado en este proyecto. Ya

sea por requerir configuraciones adicionales, formatos específicos o una interfaz gráfica poco intuitiva, el proceso de ocultamiento y recuperación de datos resultó menos directo.

Si bien algunas de estas herramientas cuentan con opciones avanzadas, como cifrado incorporado, protección por contraseña o soporte para múltiples formatos, no resultaron prácticas en su utilización considerando la perspectiva de usuario.

Steghide

Es un programa de esteganografía que oculta datos en diferentes tipos de imagen, audio y video. Soporta formatos JPEG, BMP, WAV y AU. Este programa cifra por defecto el archivo resultante, permitiendo agregar una capa extra de confidencialidad; por este motivo, además, requiere de un portador de un tamaño significativamente mayor al contenido que se desea ocultar.

Strings

Es una herramienta predeterminada de Linux que muestra por pantalla las cadenas de caracteres de un archivo.

Binwalk

Herramienta que permite buscar archivos binarios como archivos de imagen o audio embebidos en terceros. Permite la detección de datos agregados pero, por la naturaleza del cifrado, no puede siempre mostrarlos.

Casos de uso y aplicaciones actuales

La esteganografía tiene múltiples aplicaciones reales en diversos ámbitos. Desde el uso legítimo, hasta fines maliciosos. A continuación, se presentan distintos casos de uso que ilustran cómo la esteganografía se emplea actualmente en el mundo real, tanto en escenarios éticos como en situaciones que representan desafíos para la ciberseguridad.

Comunicación segura

Uno de los usos más difundidos y legítimos de la esteganografía es la transmisión segura de información sensible. Esta técnica es especialmente útil para periodistas, activistas o personas que necesitan comunicarse desde regiones con censura, vigilancia o restricciones a la libertad de expresión. La efectividad de esta técnica se incrementa con el uso de algoritmos avanzados de inserción y compresión.

Protección de derechos de autor mediante esteganografía y marca de agua oculta

Un caso concreto y documentado de uso de esteganografía enfocado en la protección de derechos de autor corresponde a la integración de técnicas de esteganografía y marcas de agua digitales en sistemas de comercio de imágenes. El objetivo de este enfoque es proteger tanto los derechos de autor como la privacidad del consumidor, ya que la marca de agua permanece invisible ante la detección casual, al mismo tiempo que permite verificar la titularidad en caso de uso sospechoso. Es un uso práctico de la esteganografía para garantizar la integridad de los archivos.

Protección de datos médicos en imágenes clínicas

Investigadores desarrollaron un modelo basado en la técnica StegaStamp para ocultar información sensible en imágenes de registros médicos, tales como radiografías o resonancias. Este enfoque integra segmentación de regiones de texto y marcas de agua ocultas que resisten ataques como recorte o compresión, logrando mantener la integridad de los datos y su calidad visual.

Espionaje gubernamental

La esteganografía ha sido una herramienta recurrente en el ámbito del espionaje desde mucho antes de la era digital. Ya a mediados del siglo XIX, con la llegada del telégrafo, individuos y empresas comenzaron a ocultar mensajes sensibles mediante técnicas esteganográficas para evitar que los operadores interceptaran información estratégica. A finales del mismo siglo, durante las campañas británicas en África, el oficial Lord Baden-Powell ocultaba mapas militares en dibujos de mariposas para no levantar sospechas si era capturado. Más adelante, en el contexto de la Primera Guerra Mundial, Alemania utilizó técnicas como los cifrados nulos, que consistían en mensajes aparentemente inocuos que, al aplicar reglas específicas (por ejemplo, tomar la segunda letra de cada palabra), revelaban contenido estratégico oculto. Con la invención del microdot, que permite incrustar páginas enteras en un punto del tamaño de un signo de puntuación, la esteganografía alcanzó una sofisticación notable. El director del FBI, J. Edgar Hoover, llegó a referirse a esta técnica como "la obra maestra del espionaje enemigo". En la actualidad, el espionaje digital continúa recurriendo a estas prácticas, aunque ahora en formato audiovisual o dentro de archivos informáticos. Las agencias de seguridad y expertos en forensia digital reconocen que la detección de contenido esteganográfico se ha convertido en un desafío clave, y recomiendan recurrir a herramientas especializadas y bases de datos forenses como la Steganography Application Fingerprint Database o la National Software Reference Library para identificar este tipo de amenazas en casos de espionaje industrial o cibercrimen.

Comunicaciones terroristas encubiertas

Desde principios de la década del 2000, agencias de inteligencia sobre el uso de esteganografía por parte de organizaciones terroristas para planificar y coordinar ataques de manera encubierta. En el Senado estadounidense se afirmó que la combinación de

cifrado y técnicas esteganográficas permitía a grupos como Al-Qaeda ocultar información crítica —como mapas, fotografías de objetivos y planes operativos— en archivos distribuidos por Internet.

Distribución de malware

El uso de canales ocultos mediante esteganografía para la comunicación maliciosa, conocido como stegomalware, ha crecido considerablemente en los últimos años. Este tipo de malware se caracteriza por ocultar su carga útil (payload) dentro de archivos aparentemente inocuos, como imágenes PNG, con el fin de evadir la detección por parte de sistemas de seguridad tradicionales. Uno de los casos estudiados es el malware IcedID, que oculta código cifrado dentro del segmento IDAT de imágenes PNG, utilizando la clave RC4 incrustada para desencriptar y ejecutar el shellcode malicioso una vez descargado en el sistema de la víctima. Otro ejemplo, StegoLoader, empleado por el grupo APT32, usa técnicas de LSB esteganográfico junto con cifrado AES para ocultar y desplegar código malicioso en etapas posteriores de un ataque.

Para abordar esta amenaza, Vielhauer et al. (2025) propusieron un modelo conceptual que combina análisis de código, comunicación y esteganálisis, permitiendo a analistas forenses y especialistas en seguridad reconstruir incidentes a partir de muestras de comunicación sospechosas. Este enfoque promueve la creación de una base de conocimiento con propiedades medibles que permiten identificar y clasificar muestras de stegomalware sin necesidad de acceder directamente al código malicioso.

Stegosploit

Uno de los ejemplos más avanzados del uso ofensivo de la esteganografía es Stegosploit, una técnica que combina ocultamiento de información con explotación de vulnerabilidades del navegador. A diferencia del malware tradicional, que suele requerir la descarga o ejecución de archivos, Stegosploit permite insertar código malicioso directamente dentro de una imagen, haciendo que su simple visualización (por ejemplo, al cargar una página web o recibir un mensaje) desencadene la ejecución del exploit en el navegador de la víctima.

El funcionamiento de Stegosploit se basa en ocultar fragmentos de código JavaScript dentro de los píxeles de una imagen usando técnicas de esteganografía (como LSB). Luego, una página web especialmente diseñada contiene código que recupera y ejecuta ese contenido oculto. Esto puede lograrse mediante llamadas a funciones estándar de HTML5 como canvas, que permiten acceder a los valores de color de cada píxel. De esta manera, el navegador decodifica y ejecuta el código sin que el usuario realice ninguna acción explícita.

- Evasión: al estar incrustado en una imagen, el código malicioso pasa desapercibido por herramientas de análisis estático y motores antivirus tradicionales.
- Ejecución sin interacción: el payload se activa automáticamente cuando la imagen es procesada por el navegador, lo que facilita ataques dirigidos o en masa.

Si bien Stegosploit fue originalmente presentado como una prueba de concepto, demuestra cómo las técnicas esteganográficas pueden evolucionar para integrarse con mecanismos de explotación, haciendo aún más difusa la frontera entre lo visible y lo malicioso.

Consideraciones ético-legales

No obstante su importancia en términos de seguridad y privacidad, el uso de técnicas esteganográficas plantea también importantes consideraciones éticas. Por un lado, la capacidad de ocultar información puede ser utilizada con fines legítimos para proteger la privacidad individual y garantizar la libertad de expresión. Por otro lado, estas mismas técnicas pueden ser aprovechadas por actores maliciosos para actividades ilícitas, como el espionaje, la distribución encubierta de malware o el intercambio secreto de información con fines delictivos.

En este sentido, resulta clave abordar la esteganografía desde una perspectiva ética, promoviendo su uso responsable y estableciendo medidas de control y regulación que prevengan abusos sin limitar innecesariamente sus aplicaciones legítimas.

Desde el punto de vista legal, el uso de técnicas esteganográficas puede generar múltiples controversias. La posibilidad de ocultar la existencia de información implica desafíos significativos para las fuerzas de seguridad y autoridades judiciales, especialmente en contextos donde la vigilancia y control del tráfico de datos es legalmente necesaria para prevenir o perseguir actividades criminales.

En muchos países, el uso legítimo de estas técnicas es considerado válido y legal siempre que se respeten las normativas vigentes relativas a la privacidad, protección de datos y seguridad informática. No obstante, cuando se emplea la esteganografía con fines maliciosos o criminales, puede conllevar sanciones legales severas, incluyendo cargos penales por espionaje, violación de privacidad, fraude o terrorismo.

En este contexto, es esencial que quienes utilicen estas técnicas estén informados de la legislación aplicable en su jurisdicción y sean conscientes de los riesgos legales derivados de su uso indebido. Asimismo, desde una perspectiva regulatoria, se hace cada vez más necesario desarrollar marcos legales específicos que clarifiquen el uso legítimo y seguro de la esteganografía, protegiendo a la sociedad de sus abusos, sin socavar las libertades fundamentales.

Conclusión

La esteganografía representa una poderosa herramienta dentro del campo de la seguridad de la información, con aplicaciones legítimas que van desde la protección de datos sensibles hasta la comunicación en contextos hostiles. Su principal ventaja radica en que no solo protege el contenido del mensaje, como ocurre con la criptografía, sino que además oculta la existencia misma de la comunicación. Esto la convierte en una técnica valiosa para complementar otros métodos de protección, ya que puede utilizarse junto con criptografía para añadir una capa extra de seguridad.

Sin embargo, también presenta limitaciones. Su capacidad de ocultamiento depende del medio utilizado, y su eficacia puede verse comprometida por técnicas modernas de análisis forense, compresión automática o transformaciones en redes sociales. Además, el uso irresponsable o malicioso, como en casos de malware y espionaje, plantea importantes desafíos éticos. De allí la necesidad de promover un uso responsable de estas técnicas, tanto en el ámbito profesional como en la investigación.

Desde el punto de vista académico y práctico, este trabajo permitió comprender no solo las bases teóricas y técnicas de la esteganografía, sino también su alcance real y las dificultades que presenta su detección. En la práctica, su capacidad de pasar inadvertida depende del tipo de archivo portador, del método de ocultamiento, del canal por el que se transmite y del nivel de sofisticación del atacante o defensor. Herramientas como StegExpose o StegSecret permiten detectar patrones sospechosos, pero no garantizan una detección completa sin intervención experta.

En definitiva, el análisis de casos reales confirma que la esteganografía es un campo en constante evolución. El surgimiento de nuevas técnicas plantea desafíos cada vez más complejos, lo que subraya la necesidad de continuar investigando y profundizando en sus aplicaciones, riesgos y mecanismos de detección.

Repository

<https://github.com/ludmilapolygin/ProyectoSeguridad>

Referencias y recursos utilizados

1. A. Fadheli, "Steganography: How to Hide Data in Images in Python - The Python Code," [thepythoncode.com](https://thepythoncode.com/article/hide-secret-data-in-images-using-steganography-python). Consultado: 2 de julio de 2025.
2. N. Meghanathan, "Steganography." Disponible: <https://www.jsums.edu/nmeghanathan/files/2015/05/CSC439-Sp2013-10-Steganography.pdf>. Consultado: 2 de julio de 2025.
3. Material de clase - Seguridad en Sistemas. Profesor Lic. Leonardo de Matteis. *Herramientas: criptografía*. Departamento de Ciencias e Ingeniería de la Computación. Universidad Nacional del Sur.
4. "Selector de color RGB," rgbcolorpicker.com. <https://rgbcolorpicker.com/>. Consultado: 2 de julio de 2025.
5. "Universidad de Buenos Aires." Disponible: http://bibliotecadigital.econ.uba.ar/download/tpos/1502-1765_SanchezArteagaJM.pdf. Consultado: 2 de julio de 2025.
6. Material de clase - Sistemas Operativos y Distribuidos. Profesor Lic. Gustavo Distel. *Memoria principal*. Departamento de Ciencias e Ingeniería de la Computación. Universidad Nacional del Sur.

7. Wannida Sae-Tang and Adisorn Sirikham, "Image Steganography-based Copyright and Privacy-Protected Image Trading Systems," *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, vol. 17, no. 3, pp. 358–375, Aug. 2023, doi: <https://doi.org/10.37936/ecti-cit.2023173.252500>.
8. D. Grover, "Steganography for identifying ownership of copyright," *Computer Law & Security Review*, vol. 14, no. 2, pp. 121–122, Mar. 1998, doi: [https://doi.org/10.1016/s0267-3649\(97\)82141-3](https://doi.org/10.1016/s0267-3649(97)82141-3).
9. Hua, C., Wu, Y., Shi, Y., Hu, M., Xie, R., Zhai, G., & Zhang, X. P. (2023). Steganography for medical record image. *Computers in biology and medicine*, 165, 107344. <https://doi.org/10.1016/j.combiomed.2023.107344>
10. A. Nag, "Low-Tech Steganography for Covert Operations," *International Journal of Mathematical Sciences and Computing*, vol. 5, no. 1, pp. 18–30, Jan. 2019, doi: <https://doi.org/10.5815/ijmsc.2019.01.02>.
11. S. Murphy, "Steganography-the New Intelligence Threat EWS 2004 Subject Area Intelligence." Disponible: <https://apps.dtic.mil/sti/tr/pdf/ADA520517.pdf>. Consultado: 5 de julio de 2025.
12. M. Warkentin, E. Bekkering, and M. Schmidt, "Steganography: Forensic, Security, and Legal Issues," *Journal of Digital Forensics, Security and Law*, vol. 3, no. 2, 2008, doi: <https://doi.org/10.15394/jdfsl.2008.1039>.
13. Vielhauer, C., Loewe, F., & Pilgermann, M. (2025). *Towards modeling hidden & steganographic malware communication based on images*. In *Proceedings of the 2025 ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '25)* (pp. 52–63). Association for Computing Machinery. doi: <https://doi.org/10.1145/3733102.3733152>
14. J. Fridrich, "Esteganálisis," Academic Press, 2007, págs. 349–381. doi: <https://doi.org/10.1016/B978-012369476-8/50016-6>. Consultado: 8 de julio de 2025.
15. J. De, T. Ahmad, and F. Han, "Comprehensive Survey on Image Steganalysis Using Deep Learning," *Array*, pp. 100353–100353, Jun. 2024, doi: <https://doi.org/10.1016/j.array.2024.100353>. Consultado: 8 de julio de 2025.
16. Kali, "Kali Tools | Kali Linux Tools," *Kali Linux*. <https://www.kali.org/tools/>. Consultado: 8 de julio de 2025.
17. L. Caviglione and W. Mazurczyk, "Never Mind the Malware, Here's the Stegomalware," *IEEE Security & Privacy*, vol. 20, no. 5, pp. 101–106, Sep. 2022, doi: <https://doi.org/10.1109/msec.2022.3178205>. Consultado: 8 de julio de 2025.