Criptografía y Seguridad

Trabajo Práctico de Implementación - 2021Q1

Secreto Compartido con Esteganografía

Grupo 19
- de la Torre, Nicolás - 58546
- Oliva, Juan Martín - 58664
- Vindis, Davor - 58663

ÍNDICE

Introducción

En este trabajo práctico se buscó implementar la encriptación de una imagen repartida entre N imágenes portadoras la cual una vez escondido del secreto dentro de ellas, las imágenes portadoras quedan exactamente igual **a la vista** pero con un mensaje oculto. Este método de ocultar mensajes dentro de otros se denomina ***esteganografía***, procura ocultar mensajes dentro de otros objetos y de esta forma establecer un canal encubierto de comunicación, de modo que el propio acto de la comunicación pase inadvertido para observadores que tienen acceso a ese canal.

Para la decodificación de la imagen se pueden tomar K de las N imágenes portadoras y descodificar para obtener la imagen secreta.

Para poder lograr cumplir con estas premisas se utilizó el ***Secreto de Shamir*** para tomar las imágenes, se utilizó ESBS para la encriptación y los métodos matemáticos de lagrange y galois en la hora de desencriptar.

1. Aspectos relativos al documento

La organización del documento nos pareció correcta, la distribución entre los sectores de resumen, introducción, esquema propuesto, resultados experimentales y conclusiones nos pareció amigable y sencilla de seguir.

El resumen es muy completo y con pocas líneas es capaz de describir con mucha precisión el trabajo práctico.

La introducción en principio parece un poco extensa pero luego de realizar el trabajo, resolver conflictos y sumergirnos en el problema, leerlo nuevamente nos genera una visión de cómo se llegó a este método de encriptación y podemos entender mejor los demás sistemas planteados, sus pros y sus contras.

La descripción de los algoritmos presentes en la sección de “esquema propuesto”, ambos pero especialmente el de recuperación nos parecieron un tanto difíciles de seguir, la notación no nos pareció del todo clara sino en sí confusa pero con la documentación extra presentada por la cátedra sumado a material que se puede encontrar en línea es finalmente entendible y de ayuda.

En la siguiente imagen se detalla un error de notación donde el último término del polinomio debiera ser de grado ‘k-1’ pero se ausenta él “-“.

$$F(x_i) = S + a_1 x_i + a_2 x_i^2 + \ldots + a_{k\ 1} x_i^{k\ 1}, \qquad i = 1 \ldots n$$

Fig 1: Fórmula (1) del documento “Sistema de Imagen Secreta Compartida con Optimización de la Carga Útil”

Las imágenes y gráficos presentes a lo largo del documento dieron un aporte sustancial para facilitar el entendimiento de pasos y conceptos.

2. Carga útil

2.1 ¿A qué se refiere?

Cuando hablamos de carga útil en el concepto de este Trabajo Práctico nos referimos a aquellos datos que contengan el mensaje real con el cual estamos trabajando.

Dentro del documento encontramos que el esquema propuesto utiliza específicamente una **Interpolación de Lagrange** que se resuelve de manera iterativa bajo un campo GF(2^8) esto no es coincidencia y la razón es que comparado con otros métodos propuestos, este presenta mayor carga útil.

2.2 ¿Qué relación existe entre k y el tamaño de la portadora?

K es un factor clave ya que además de ser parte de la definición del esquema de umbral (k, n) que utilizamos, incrementar su valor implica incrementar la carga útil como lo demuestra la fórmula :

$$Payload = \frac{(NPixeles * NColor * K)}{4}$$

Por otro lado hay que tener en cuenta que existe una relación clave entre K y el tamaño de la imagen portadora (pixeles) y es que esta última debe ser divisible por K para poder realizar una correcta división y distribución de bloques

3. Ventajas y desventajas que ofrece trabajar en GF(2^8) respecto de trabajar con congruencias módulo

Dentro de las **ventanas** de trabajar con GF(2^8) la principal de ellas es que **GARANTIZA** una recuperación sin pérdidas de la imagen secreta. También como se mencionó anteriormente permite una mayor carga útil lo que resulta en imágenes de mayor calidad.

Dentro de las **desventajas** que encontramos fue un incremento de complejidad al tener que definir las operaciones que utilizamos, dentro del campo de Galois y también implica mayor uso de espacio para definir las tablas necesarias.

4. ¿Se puede trabajar con otro polinomio generador? ¿podría guardarse como “clave”?

Si, se puede trabajar con otro polinomio mientras cumpla las siguientes premisas:

- Debe ser un polinomio irreducible de grado N
- Debe tener un número impar de términos
- Debe tener por lo menos una constante

Este polinomio se puede considerar como clave ya que es necesario y define cómo distribuir el secreto.

5. ¿Por qué se pueden guardar secretos de todo tipo? Por ejemplo: imágenes, pdf, ejecutables

Así mismo como se contestó en la pregunta 3 se pueden guardar secretos de todo tipo gracias a que trabajamos con $GF(2^8)$ y explotamos su propiedad de recuperación de datos del secreto sin pérdidas de los mismos.

Esto garantiza que no se van a perder bits de información, estos bits pueden ser de una imagen, un archivo o un ejecutable.

6. ¿Cómo podría adaptarse la implementación realizada para poder guardar un archivo de imagen completo? (guardando encabezado y pixeles)

Para poder guardar un archivo de imagen completo, incluyendo encabezado y pixeles se debería adaptar el trabajo que ya tenemos hecho de la siguiente manera.

Primero debemos tener en cuenta los headers de las imágenes y no obviarlos cómo lo veníamos haciendo, esta información extra haría que se necesiten imágenes sombra de mayor tamaño para poder cubrir toda la data nueva que tenemos en el secreto.

Finalmente a la hora de recuperar el secreto tenemos que tener en cuenta el header y su tamaño para diferenciarlo de la imagen

7. Analizar cómo resultaría el algoritmo si se usaran imágenes en color (24 bits por píxel)

El algoritmo funciona de manera muy similar y seguiría teniendo todas las propiedades que tiene hasta el momento . El cambio que se debería hacer sería el de computar los colores por lo cual necesito representar los píxeles con el sistema RGB, ya no alcanza con decir si es blanco o negro.

Por lo tanto se necesitaría el triple de bits por pixel sumando un total de 24 bits por pixel. El incremento de este valor se vería reflejado probablemente en mayores tiempos de ejecución pero no da el valor agregado de poder tener imágenes de buena definición en color.

8. Explicar si se podrían tomar los bloques de otra manera en lugar de matrices de 2x2.

Se podrían tomar bloques de otra manera que no sean de distribución 2x2 siempre y cuando se respete la distribución de bloques tanto en el proceso de encriptado como durante el proceso de desencriptado.

También hay que tener en cuenta el formato de la imagen, es decir, si tomamos distribución de bloques 2x2 estamos obligados a tener cantidad par de filas de píxeles.

En conclusión sí se podrían tomar bloques de otra manera por ejemplo 1x4 en lugar de matrices 2x2.

9. Aspectos relativos al algoritmo implementado

9.1. Facilidad de implementación

Gracias al material provisto por la cátedra, los papers y las clases dictadas sobre **Galois** y **Secreto de Shamir**, y a la cantidad de documentación existente en internet sobre todos los temas planteados en el Trabajo Práctico, la implementación no tuvo mayores inconvenientes “técnicos”.

Además utilizamos la librería ***dirent.h*** que fue de mucha ayuda para ver la cantidad de imágenes/archivos que había en el directorio enviado.

Los mayores inconvenientes encontrados estuvieron relacionados a la verificación del correcto funcionamiento de los distintos módulos o partes del programa (los algoritmos matemáticos, el procesado de las imágenes Bitmap, el pasaje matricial de las

imágenes), un pequeño error en alguna de estas partes a menudo resultó difícil de detectar.

A pesar de utilizar una correcta división de tareas, y encarar el proyecto de la manera recomendada por la cátedra, al integrar todos los módulos encontramos problemas que no habían sido contemplados.

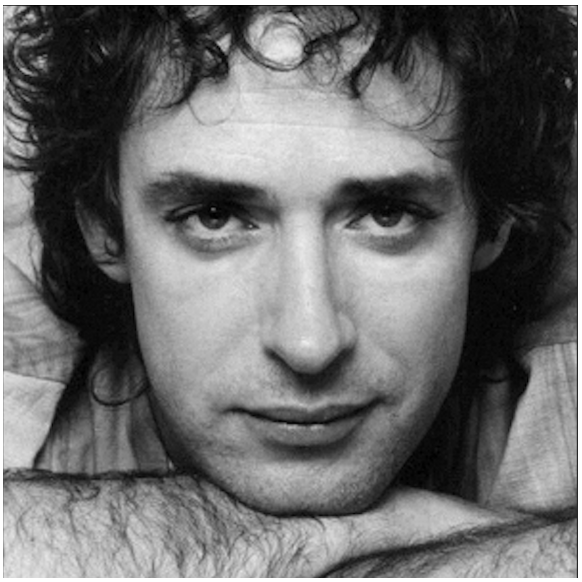
10. ¿En qué situaciones aplicarían este tipo de algoritmos?

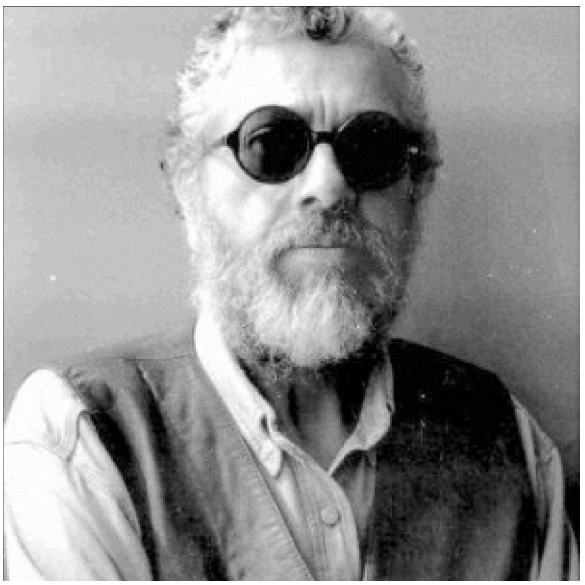
Se podría usar en alguna aplicación donde se reparte la clave para un grupo de personas y cierta acción se habilita en cuanto K personas mínimo presenten su key. De esta manera se puede restringir el acceso a un recurso para ser usado bajo supervisión de un grupo.

Así mismo cualquier sistema donde no se quiera centralizar una llave en un único lugar sino que se quiera repartir entre varias partes para mitigar las chances de que pueda ser robada.

Imágenes proporcionadas

En el siguiente apartado podrán ver las imágenes utilizadas para ser codificadas y decodificadas en ese tp, son las mismas imágenes provistas por la cátedra.





Secreto guardado en las imágenes enviadas

Las cuatro imágenes mostradas anteriormente contenían un secreto oculto, que se muestra a continuación.