

Recuperación de la imagen usando la llave "cifrados_2025":



Observaciones:

- No fue necesario convertir la imagen a base64 para recuperarla, ya que el XOR se aplica directamente sobre los bytes de la imagen sin necesidad de una transformación intermedia.
- Se aplicó XOR directamente a los bytes de la imagen utilizando una clave de texto.
- La clave de texto se reutiliza de manera cíclica para garantizar que se pueda aplicar correctamente sobre toda la imagen (usando el operador %...)

¿Por qué una Imagen se Corrompe al Aplicar XOR con una Llave de Texto?

Longitud de la clave: Si la clave es más corta que los datos de la imagen, se reutiliza cíclicamente, lo que puede introducir patrones repetitivos y degradar la calidad visual.

Formato de la imagen: Las imágenes pueden estar en diferentes formatos (RGB, RGBA, escala de grises, etc.). Si la clave no se adapta al formato específico, la operación XOR puede producir resultados inesperados o corrupción de datos.

Integridad de los datos: Las imágenes contienen metadatos y estructuras de encabezado esenciales para su correcta interpretación. Aplicar XOR indiscriminadamente a todos los bytes, incluidos estos metadatos, puede corromper la estructura de la imagen.

Compatibilidad de caracteres: Las claves de texto están compuestas por caracteres que, al representarse en bytes, pueden no abarcar todo el rango de valores posibles (0-255).

Aplicación de XOR a Dos Imágenes



Inconvenientes y Soluciones

- Se observó que el resultado depende en gran medida de la similitud en la estructura de los píxeles entre ambas imágenes. El método `resize` con remuestreo `LANCZOS`, implementado en la librería Pillow, permite redimensionar imágenes preservando detalles y suavizando bordes, lo que ayuda a evitar distorsiones al aplicar XOR entre imágenes de distintos tamaños.
- La conversión a RGB ayudó a evitar errores de formato, asegurando que ambas imágenes fueran compatibles en términos de datos de píxeles.