Tabla de contenido

1. Resolución 2284 de 2023

Según la resolución 2284 del 28 de diciembre de 2023 y la resolución 1885 del 30 de septiembre de 2024, se ha establecido lo siguiente bajo el anexo técnico No. 2 Envío y recepción de los soportes de cobro, numeral 1.1:

1.1 Enviar los archivos de los soportes encriptados, con clave asimétrica tipo RSA válida tanto para cifrar como para firmar digitalmente, en coherencia con la normativa sobre la seguridad de la información, derecho fundamental a la intimidad y al Habeas Data.

2. ¿Qué es Cifrado RSA?

Tomado de *https://ciberseguridad.com/guias/prevencion-proteccion/criptografia/cifrado-rsa/#%C2%BFQue_es_el_cifrado_RSA*

Bajo el cifrado RSA, los mensajes se cifran con un código llamado clave pública, que se puede compartir abiertamente. Debido a algunas propiedades matemáticas distintas del algoritmo RSA, una vez que se ha activado un mensaje con la clave pública, solo se puede descifrar con otra clave, conocida como clave privada. Cada usuario de RSA tiene un par de claves que consta de sus claves públicas y privadas. Como sugiere el nombre, la clave privada debe mantenerse en secreto.

3. ¿Dónde se usa?

Tomado de *https://ciberseguridad.com/guias/prevencion-proteccion/criptografia/cifrado-rsa/#%C2%BFDonde_se_usa*

El cifrado RSA se usa a menudo en combinación con otros esquemas de cifrado o para firmas digitales que pueden probar la autenticidad e integridad de un mensaje. Por lo general, no se usa para cifrar mensajes o archivos completos, porque es menos eficiente y consume más recursos que el cifrado de clave simétrica.

Para hacer las cosas más eficientes, un archivo generalmente se encriptará con un algoritmo de clave simétrica y luego la clave simétrica se encriptará con encriptación RSA. Bajo este proceso, solo una entidad que tenga acceso a la clave privada RSA podrá descifrar la clave simétrica.

Sin poder acceder a la clave simétrica, el archivo original no se puede descifrar. Este método se puede utilizar para mantener seguros los mensajes y los archivos, sin tardar demasiado ni consumir demasiados recursos informáticos.

4. Cifrado AES vs RSA

AES a diferencia de RSA, es cifrado mediante una única clave conocida entre emisor y receptor. A continuación, se remiten diferencias entre ambas metodologías de encriptación.

Tomado de https://www.ssldragon.com/es/blog/rsa-aes-cifrado/

Visión general de RSA frente a AES

	Cifrado RSA	Cifrado AES
Tipo	Asimétrico	Simétrico
Longitud de la llave	2048, 3072 ó 4096 bits	128, 192 o 256 bits
Velocidad	Más lento, alto coste computacional	Más rápido, eficiente para grandes datos
Casos de uso común	Firmas digitales, SSL/TLS	Almacenamiento de datos, VPN, seguridad inalámbrica

5. Selección de la metodología de encriptación

Para el cifrado de soportes asociados a una cuenta médica de facturación, se ha seleccionado la metodología híbrida, es decir, a través de AES y RSA se dará cumplimiento a lo requerido por la resolución 2284 de 2023. Es importante resaltar que la selección híbrida se genera porque RSA no permite la encriptación de archivos de tamaños grandes, y AES sí lo permite, por tanto, será necesario como lo indica el numeral 3 cifrar los archivos mediante AES y posterior a esto cifrar la clave usada de AES mediante RSA con la llave pública otorgada.

6. Pasos para la encriptación del soporte

A continuación, se nombra paso a paso de lo que se debe realizar para la generación de encriptación de los documentos soporte:

1. Selección del archivo a cifrar

- **Interfaz de usuario para seleccionar archivo**: El usuario debe poder seleccionar un archivo desde su sistema de archivos local. Este archivo es el que se va a cifrar.
- **Obtener la ruta y extensión**: Una vez seleccionado el archivo, se debe capturar la ruta completa del archivo y obtener su extensión (por ejemplo, .txt, .jpg, .pdf), ya que esa información será útil más adelante.

2. Leer el contenido del archivo

- **Leer el archivo como datos binarios**: El archivo debe leerse completamente como un arreglo de bytes. Los bytes representan el contenido del archivo que será cifrado.

3. Cifrado del archivo con un algoritmo simétrico (AES)

- **Generar una clave y un vector de inicialización (IV)**: Se genera una clave aleatoria que se utilizará para cifrar el archivo. Además, se genera un vector de inicialización (IV) único para cada archivo, que ayuda a asegurar que los mismos datos no siempre se cifren de la misma manera.
- **Cifrar el archivo**: Usando la clave y el IV generados, se cifra el contenido del archivo. Este proceso transforma el archivo original en un conjunto de datos cifrados que no es legible sin la clave correcta.

4. Cifrado de la clave AES con un algoritmo asimétrico (RSA)

- **Cargar una clave pública RSA**: Para proteger la clave utilizada en el cifrado simétrico (AES), se emplea un algoritmo de cifrado asimétrico, como RSA. Para esto, se utiliza una clave pública RSA. La cual debe ser la siguiente**: (NO debe ser compartida y debe mantenerse con todos los criterios de seguridad, confidencialidad, integridad y disponibilidad)**

<RSAKeyValue><Modulus>7C8b4ybsYSc/2CuUgwjyRbDfgTr0THrKnfFHqZBdPqirA9uMiTCPri0QeyQr
ViqrWTBiGY6tbp1NZDzRt271xcEHAWVCMnP4Y+Fv17utLLsT8fLkOsa6bsFiAmTaAufrC/A3YBQwaIaL
nojaDGD4CQCXrJla4b+wnZRL34EuK9DgQcNbUGHfcPhE52MdnifFIS/7l+AxEVlSWXZ+iO5L0vZEU1nq
ZMnusyon/bD3wGoJlsZXJmZc9kLKl6qXrHbbIFWHF5VQNHjgu7vMhgj8fnZgAy544jYZAkKwvChgn4M
pSJ2TGguRjNqJKPcbs/r0SBMZNjzM6Pk+AXrWx5LovJFuPPiXFzPmwbqlPlevj/mLHz+md1nCxD80WUc
ZbN1GLPyjrorBJ4l5Z+dEJ/UZ7oaKpnaLNtfhZkgJPg3V4BvOqWhD883XeMCt5iXYqN7DDtNJwGO2jBA
ACp2zoYuvXvSeaUNZzk2veKLfsxGQUT8DkvUM3HtPO6o0LwyY5CHRujaYZH/J++Ka1ne3qJ4m9KHuc
au6xwf1TB91cjNcXdyV85+urs3oDIi0Vw7igWfGbrK97r3f5dzUx/G6B9gjTgjzfT+Q7b13BvT6yEbZeVJh
YG8XkKZR1foCBDSqKet9pDVYPzzpve7fVYmqs85CFjIm1e3rDW4NO8SIqQUgCrk=</Modulus><Expo
nent>AQAB</Exponent></RSAKeyValue>

- **Cifrar la clave AES con RSA**: La clave que se generó para el cifrado del archivo (la clave AES) se cifra con la clave pública RSA. Esto asegura que solo quien tenga la clave privada correspondiente pueda descifrar la clave AES y, por ende, el archivo cifrado.

5. Guardar el archivo y la clave cifrados

- **Guardar el archivo cifrado**: El archivo cifrado se guarda en el sistema de archivos local. Antes de guardar los datos cifrados, es recomendable incluir la extensión original del archivo para poder restaurarla cuando sea necesario. Esto puede hacerse guardando primero la longitud de la extensión, luego la extensión en sí misma y, finalmente, el archivo cifrado junto con el IV generado.
- **Guardar la clave cifrada por separado**: La clave que se usó para cifrar el archivo (AES) se guarda de forma cifrada en un archivo separado ".key". Este archivo contiene la clave cifrada utilizando RSA. Debe tener el mismo nombre del archivo cifrado.

6. Guardar longitud de la extensión y la extensión

La idea es guardar la extensión del archivo original dentro del archivo cifrado para poder recuperarla cuando sea necesario. Para ello, se usa una técnica en la que primero se almacena el tamaño de la extensión y luego la extensión en sí misma. Esta información se guarda al principio del archivo cifrado. Esto porque al guardar la longitud de la extensión primero, se puede saber cuántos bytes ocupará la extensión cuando se lea el archivo cifrado, sin tener que hacer suposiciones. Y Luego, almacenar la extensión en sí misma permite que, cuando se descifre el archivo, se pueda recuperar el nombre original del archivo, restaurando su formato (por ejemplo, .jpg, .txt, .pdf, etc.).

Paso 1: Calcular la Longitud de la Extensión

La longitud de la extensión es simplemente el número de caracteres que tiene la extensión del archivo, incluida el punto (.). Por ejemplo:

- Para un archivo imagen.jpg, la extensión es .jpg, que tiene 4 caracteres.
- Para un archivo documento.pdf, la extensión es .pdf, que tiene 4 caracteres.

Paso 2: Convertir la Longitud en Bytes

Una vez que se tiene la longitud de la extensión, se debe convertir a una representación en bytes. En la mayoría de los lenguajes de programación, la longitud de un entero (que representa la longitud de la extensión) se guarda en 4 bytes.

Paso 3: Escribir la Longitud y la Extensión en el Archivo Cifrado

Para almacenar la longitud de la extensión y la extensión en sí misma en el archivo cifrado, se siguen estos pasos:

1. Escribir la longitud (en bytes) de la extensión en los primeros 4 bytes del archivo.
2. Escribir los bytes de la extensión (por ejemplo, los caracteres de .jpg o .pdf).

   Al hacerlo, nos aseguramos de que, al leer el archivo cifrado, primero podamos saber cuántos bytes corresponderán a la extensión (usando los primeros 4 bytes) y luego leer los bytes correspondientes a la extensión del archivo.

Flujo Detallado

1. Obtener la extensión del archivo original: Por ejemplo, si el archivo es imagen.jpg, se obtiene .jpg.
2. Calcular la longitud de la extensión: En este caso, la longitud de .jpg es 4.
3. Convertir la longitud a bytes: Usando 4 bytes para representar el valor 4.
4. Convertir la extensión a bytes: Los caracteres de la extensión .jpg se convierten en su representación de bytes. Por ejemplo, la extensión .jpg en UTF-8 sería:
   - . → byte 46 (en ASCII)
   - j → byte 106
   - p → byte 112
   - g → byte 103

   Entonces, la extensión .jpg se representaría como el arreglo de bytes [46, 106, 112, 103].
5. Escribir la longitud y la extensión en el archivo:
   - Primero, se escribe la longitud de la extensión (4 bytes) al principio del archivo cifrado.
   - Luego, se escriben los bytes que representan la extensión.

7. Resumen de la Metodología

1. **Selección del archivo**: Permitir que el usuario seleccione un archivo desde su sistema de archivos.
2. **Lectura del archivo**: Leer el archivo seleccionado como datos binarios (arreglo de bytes).
3. **Cifrado del archivo con AES**: Generar una clave y un IV, luego cifrar el archivo utilizando AES.
4. **Cifrado de la clave AES con RSA**: Cifrar la clave AES generada usando una clave pública RSA.
5. **Guardar el archivo cifrado**: Almacenar el archivo cifrado junto con la extensión original y el IV. El archivo tendrá el nombre y extensión ".enc".
6. **Guardar la clave cifrada**: Almacenar la clave AES cifrada por separado en un archivo. El archivo tendrá el mismo nombre del archivo cifrado terminado en ".enc.key"