Programación de Servicios y Procesos

TÉNICAS DE PROGRAMACIÓN SEGURA

Cifrado y descifrado de datos utilizando el algoritmo AES en Java

Ismael Vega Hormigos

DA2D1A

Contenido

Contexto

El cifrado de datos es fundamental para garantizar la confidencialidad de la información, transformando datos legibles en datos ininteligible mediante el uso de algoritmos y claves. Con diferentes mecanismos se busca proteger los datos sensibles, tanto almacenados como en tránsito.

AES (Advanced Encryption Standard) es uno de los métodos mas utilizados para cifrar datos. Este algoritmo simétrico, adoptado como estándar por el Instituto Nacional de Estándares y Tecnología (NIST) en 2001, utiliza una misma clave para cifrar y descifrar los datos. AES ofrece tres tamaños de clave: 128, 192 y 256 bits, donde una clave más larga proporciona un mayor nivel de seguridad.

AES no se aplica directamente, si no que se implementa junto con modos de operación que definen como procesar bloques de datos. Los mas comunes son:

- ECB (Electronic Codebook): Cifra cada bloque de datos de la misma forma. Es considerado inseguro. Requiere relleno.
- CBC (Cipher Block Chaining): Introduce aleatoriedad al usar un vector de inicialización (IV) para que cada bloque dependa del anterior Proporcionando mayor seguridad. Requiere relleno y permite descifrar en paralelo.
- CFB (Cipher Feedback) y OFB (Output Feedback): Convierte AES en un flujo de cifrado, procesando los datos como un flujo continuo en lugar de bloques fijos.
- CTR (Counter): Ofrece cifrado y autenticación simultáneamente, verificando tanto la integridad como la confidencialidad de los datos. Permite cifrado y descifrado en paralelo, así como cifrar un flujo.

Explicación del programa

El programa consiste en un cifrado y descifrado de texto introducido por la consola utilizando de forma simple el algoritmo de cifrado simétrico AES (Advanced Encryption Standard) aplicando el modo de operación ECB (Electronic Codebook).

El programa está estructurado en dos clase:

La primera denominada *“ClaveAES”* que cuenta con tres métodos estáticos.

1. Generar una clave secreta de 256 bits que será utilizada tanto para el cifrado como el descifrado del texto
2. Cifrar un texto pasado como parámetro (junto con la clave generada). Transformaremos el texto a un array de bytes para su cifrado y lo codificaremos en formato Base64 para que sea legible

3. Descifrar el texto previamente cifrado pasado como parámetro (de nuevo junto con la clave secreta). Decodificaremos el texto en Base64, aplicaremos el descifrado y recuperaremos el texto original.

La segunda es será la clase principal, en la que primero generaremos la clave secreta y la mostraremos al usuario por consola codificada en Base64. Seguidamente le pediremos al usuario introducir un texto mediante la consola recuperándolo mediante la clase Scanner. Posteriormente procederemos al cifrado de datos mostrándolo por consola al usuario y por último descifraremos el texto para mostrarlo de nuevo por consola. El programa captura las excepciones chequeadas lanzadas en los métodos de la clase *"ClaveAES"* con un bloque try/catch con recursos para optimizar el cierre del objeto scanner.

Código fuente

Adjuntado en la entrega junto con el jar

Repositorio en GitHub

https://github.com/Vladerking/cifrado_datos_ivega.git

Bibliotecas Usadas

Javax.crypto → Proporciona funcionalidades de cifrado, descifrado, generación de claves, etc.

Java.util → Proporciona utilidades, en este programa concreto hemos utilizado la clase Base64 para la codificación y decodificación del texto y la clase Scanner.

Dependencias

No hemos usado dependencias en este programa, todas las clases utilizadas forman parte de la biblioteca estándar de Java incluidas en el JDK.

Bibliografía

- Baeldung. (s.f.). Java AES Encryption and Decryption https://www.baeldung.com/java-aes-encryption-decryption

- HowToDoInJava. (s.f.). AES 256 Encryption and Decryption https://howtodoinjava.com/java/java-security/aes-256-encryption-decryption/

- Cernera, G. (s.f.). Encrypting and Decrypting a Message Using Symmetric Keys with Java Explained Step by Step https://gregorycernera.medium.com/encrypting-and-decrypting-a-message-using-symmetric-keys-with-java-explained-step-by-step-with-a523b67877d8

- Sirohi, D. (s.f.). Java AES Encryption and Decryption. Recuperado de https://medium.com/@deepak.sirohi9188/java-aes-encryption-and-decryption-1b30c9a5d900

- tutorialplus. (2021). *Encrypt and Decrypt Data Using AES Algorithm in Java* https://www.youtube.com/watch?v=LtUU8Q3rgjM&ab_channel=tutorialplus