

Configuración librería encriptación

Librería para encriptar y desencriptar datos de forma segura, garantizando la protección de la información transmitida y recibida.

Uso

Para comenzar, sigue estos pasos:

- Instale la librería de npm `crypto-js` [Enlace a la librería](#)

```
npm i crypto-js
```

- Crea un archivo **.npmrc** en la raíz del proyecto, el cual va a contener la configuración para instalación de la librería alojada en Nexus Repository.
- Agregue la dependencia al archivo **package.json** en el bloque "dependencies" (**Recuerde verificar la versión actual de la librería**) e instale las dependencias.

```
"crypto-util-library":  
"http://10.1.40.130:8081/repository/npm-hosted/crypto-util-library/-/crypto-util-library-1.0.0.tgz"
```

- Crea un servicio para centralizar y gestionar la encriptación y desencriptación de datos, utilizando los métodos proporcionados por la librería **crypto-util-library**. Esto facilita la reutilización del código y garantiza un manejo seguro y consistente de la información en toda la aplicación.

```
import { CryptoUtil } from 'crypto-util-library';  
  
decryptResponse(data: any) {  
    return this.cryptoUtil.decryptObject(data);  
}  
encryptData(data: any) {  
    return this.cryptoUtil.encryptObject(data);  
}
```

- Configura un **interceptor** encargado de la encriptación y desencriptación de los datos en las solicitudes HTTP. Este interceptor recibe como parámetro un **archivo de configuración** que permite definir una lista de URLs excluidas, en las cuales no es necesario aplicar estos procesos. De esta manera, se optimiza el rendimiento y se garantiza que solo las comunicaciones sensibles sean protegidas mediante encriptación.
- [Interceptor](#)
- [Archivo de configuración](#)
- Realice la configuración del interceptor en el archivo **app.conf.ts** en el array de providers y asegúrese de pasar el archivo de configuración como parámetro para definir las reglas de encriptación y exclusión de URLs. La implementación debe realizarse de la siguiente manera::

```
import { cryptoInterceptorInterceptor } from
'./shared/interceptors/crypto-interceptor.interceptor';
import { cryptoConfig } from './shared/interceptors/crypto.config';

export const appConfig: ApplicationConfig = {
  providers: [
    provideHttpClient(
      withInterceptors([cryptoInterceptorInterceptor(cryptoConfig)])
    )
  ]
};
```

[←Regresar](#)

Documento actualizado: 03/03/2025

Elaborado por: Luis Ardila - Desarrollador frontEnd

From:
<http://wiki.adacsc.co/> - Wiki

Permanent link:
<http://wiki.adacsc.co/doku.php?id=ada:howto:sicoferp:factory:new-migracion-sicoferp:front:configuracion-libreria-encryptacion>

Last update: 2025/03/18 14:14

