Implementación de un Single Sign-On (SSO) como Proxy Inverso para Servicios API Rest

1. Introducción: ¿Qué es un Single Sign-On (SSO)

El Single Sign-On (SSO), o Inicio de Sesión Único, es un mecanismo de autenticación que permite a un usuario acceder a múltiples aplicaciones o servicios con un solo conjunto de credenciales (usuario y contraseña). En lugar de tener que iniciar sesión por separado en cada sistema, el usuario se autentica una sola vez ante un servicio central de confianza.

Aplicación en la vida real: Un ejemplo común es el ecosistema de Google. Cuando inicias sesión en tu cuenta de Gmail, automáticamente obtienes acceso a otros servicios como YouTube, Google Drive y Google Calendar sin necesidad de volver a introducir tus credenciales. La cuenta de Google actúa como el proveedor de identidad central que valida tu identidad para todos los servicios asociados.

2. Arquitectura Propuesta: SSO como Proxy Inverso

En este proyecto, el SSO no solo gestionará la autenticación, sino que también actuará como un proxy inverso. Esto significa que será el único punto de entrada para los clientes. El SSO interceptará todas las peticiones, validará la identidad del usuario y, si la validación es exitosa, redirigirá la petición al servicio API Rest interno correspondiente.

3. Seguridad con SSL/TLS

Para proteger las credenciales y los tokens de acceso que viajan por la red, toda la comunicación entre el cliente y el servidor SSO debe ser segura. Esto se logrará mediante el protocolo SSL/TLS (HTTPS).

Para que esto funcione, el servidor SSO debe tener instalado un certificado de servidor SSL. Cuando un cliente se conecta, el servidor presenta este certificado. El cliente utiliza la **llave pública** contenida en el certificado para encriptar los datos sensibles (como la contraseña). Estos datos solo pueden ser desencriptados por el servidor, ya que es el único que posee la **llave privada** correspondiente. Esto garantiza la confidencialidad e integridad de la información.

4. Descripción del Problema y Objetivos

Se requiere diseñar e implementar un servicio de Single Sign-On (SSO) que funcione como un API Rest y actúe como un proxy inverso para un conjunto de servicios API Rest existentes.

4.1. Endpoint de Autenticación:

- El SSO debe exponer un endpoint (ej. `/login`) que reciba en el cuerpo de la petición un usuario, una contraseña y una lista de los servicios a los que se desea acceder.
- Tras una validación exitosa, el SSO debe generar y devolver un token de acceso (se sugiere usar JSON Web Tokens - JWT) con un tiempo de vida definido.

4.2. Proxy Inverso y Validación de Token:

- Todas las peticiones a los servicios API Rest protegidos deben pasar a través del SSO.
- El cliente debe incluir el token de acceso de autorización (`Authorization: Bearer <token>`) en cada petición recurrente.
- El SSO debe interceptar cada petición, validar la firma y la vigencia del token.
- Si el token es válido, el SSO redirigirá la petición al servicio interno correspondiente.

- Si el token es inválido, ha expirado o no se proporciona, el SSO debe devolver una respuesta de error `401 Unauthorized`.

4.3. Seguridad:

- Toda la comunicación con el SSO debe realizarse obligatoriamente a través de HTTPS.

Se valorará la solución presentada en base a los siguientes criterios:

- Funcionalidad del Proceso de Autenticación
- Defensa de la Solución y Claridad Conceptual