DOCUMENTACIÓN: CONFIGURACIÓN DE SEGURIDAD HELMET

DESCRIPCIÓN GENERAL

Este módulo proporciona una capa básica de seguridad para la aplicación mediante la implementación de políticas de protección HTTP. Su función principal es aplicar configuraciones de seguridad esenciales a nivel de middleware para proteger la aplicación de vulnerabilidades web comunes.

FUNCIONALIDAD PRINCIPAL

El módulo utiliza la librería Helmet para configurar automáticamente varias cabeceras HTTP de seguridad que ayudan a prevenir ataques como clickjacking, sniffing de MIME type, y exposición de información sensible en las cabeceras.

ENFOQUE DE SEGURIDAD ESTRATIFICADA

La implementación adopta un enfoque de seguridad en capas donde la política de seguridad de contenido más granular y específica se delega al proxy reverso. Esto permite una gestión más flexible y centralizada de las políticas de seguridad mientras se mantienen las protecciones básicas a nivel de aplicación.

CONFIGURACIÓN ESPECÍFICA

POLÍTICA DE SEGURIDAD DE CONTENIDO (CSP)

Se desactiva intencionalmente a nivel de aplicación para permitir que el proxy reverso gestione esta política. Esto es particularmente importante para aplicaciones que integran servicios externos como Google Identity Services, que requieren ajustes específicos de nonces y hashes en la CSP.

PROTECCIONES ACTIVAS

A pesar de desactivar la CSP, Helmet continúa aplicando otras protecciones esenciales como:

Prevención de clickjacking mediante cabeceras X-Frame-Options

Control de tipo MIME para prevenir sniffing de contenido

Ocultamiento de información de tecnología subyacente en cabeceras X-Powered-By

Configuración de HSTS para forzar conexiones seguras

Prevención de inclusión de sitios en iframes

INTEGRACIÓN CON ARQUITECTURA

Este módulo se integra dentro de una arquitectura de seguridad más amplia donde diferentes capas tienen responsabilidades específicas. La aplicación maneja las

protecciones básicas, mientras que el proxy reverso se encarga de políticas más complejas que requieren ajustes frecuentes o integración con múltiples servicios.

BENEFICIOS DEL ENFOQUE

La separación de responsabilidades permite actualizaciones de políticas de seguridad sin necesidad de modificar el código de la aplicación, facilita la gestión de excepciones para servicios de terceros, y mantiene una postura de seguridad robusta mientras se mantiene la flexibilidad operativa.

IMPLEMENTACIÓN COMO MIDDLEWARE

La función de seguridad se aplica como middleware global en la aplicación, lo que significa que todas las rutas y endpoints reciben automáticamente estas protecciones sin necesidad de configuración adicional en cada controlador o ruta específica.