**Propuesta de Análisis de vulnerabilidades para el aplicativo web DocBase3 con OWASP para la empresa Diebold Nixdorf.**

Fabian Eduardo Castillo Cruz, Pedro Yulian Pabon Amaya, Favian Alejandro Tribaldos, Luis Enrique Parada Roa

Especialización en Seguridad Informática, Fundación Universitaria Compensar

Propuesta de consultoría para la empresa Diebold Nixdorf

**Miguel Ricardo Pérez Pereira**

27 de mayo de 2023

Esta obra está bajo una Licencia Creative Commons Atribución-NoComercial-CompartirIgual 4.0 Internacional.

# Propuesta de Análisis de vulnerabilidades para el aplicativo web Docbase3 con OWASP para la empresa Diebold Nixdorf.

**Fabian Eduardo Castillo Cruz, Pedro Yulian Pabon Amaya, Favian Alejandro Tribaldos, Luis Enrique Parada Roa**

**Especialista en Seguridad informática**

Director (a):
Ingeniero(a), Jairo Cesar Gómez Acero
Codirector (a):

Ingeniería y Tecnología del Software:
Seguridad informática

Fundación Universitaria Unipanamericana – Compensar
Facultad de Ingeniería, Especialización en Seguridad Informática
Bogotá D.C, Colombia
2023

**RESUMEN**

Diebold Nixdorf es una corporación estadounidense que ofrece principalmente soluciones en la producción, venta, instalación y servicio de sistemas de transacciones de autoservicio (es decir, cajeros automáticos y terminales posventa), productos de seguridad física y electrónica (como cajas fuertes y sistemas de procesamiento de divisas), software y servicios profesionales relacionados con los mercados financieros y comerciales mundiales.

La empresa Diebold Nixdorf tiene un aplicativo web llamado DocBase3 el cual recibe los incidentes de los clientes corporativos del área financiera a nivel nacional, para los cuales se tienen ANS (Acuerdos de Niveles de Servicio) que de no cumplirse conlleva a sanciones económicas. Fallos de seguridad en el aplicativo web DocBase3 ha generado pérdidas económicas significativas durante el año 2022 por incumplimiento en los ANS (Acuerdos de Niveles de Servicio).

Dado lo anterior, en este trabajo se expone una propuesta de consultoría a la empresa Diebold Nixdorf que realice un análisis de vulnerabilidades para aplicativos webs con metodologías de OWASP (Open Web Application Security Project) para identificar y remediar cualquier posible vulnerabilidad describiendo y proporcionando recursos, herramientas y directrices para ayudar a la organización a desarrollar y mantener la aplicación web DocBase3 más segura.

**Palabras clave: (DocBase3, vulnerabilidades, ANS, OWASP).**

**Abstract**

Diebold Nixdorf is an American corporation that primarily provides solutions in the production, sale, installation, and service of self-service transaction systems (ATMs and after-sales terminals), physical and electronic security products (such as safes, and data processing systems). foreign exchange), software, and professional services related to global financial and trading markets.

The Diebold Nixdorf company has a web application named DocBase3 which receives incidents from corporate clients in the financial area at a national level, for which SLA (Service Level Agreements) are in place, which if not complied with leads to economic sanctions. Security flaws in the DocBase3 web application: have generated significant economic losses during the year 2022 due to non-compliance with the SLA (Service Level Agreements).

Given the above, this paper exposes a consulting proposal to the Diebold Nixdorf company that performs a vulnerability analysis for web applications with OWASP (Open Web Application Security Project) methodologies to identify and remedy any possible vulnerability by describing and providing resources, tools and guidelines to help your organization develop and maintain the most secure DocBase3 web application.

**Keywords: (DocBase3, vulnerabilities, SLA, OWASP)**

# Contenido

**RESUMEN** ..... **V**

**INTRODUCCIÓN** ..... **11**

ANTECEDENTES Y JUSTIFICACION ..... 12
FORMULACION DEL PROBLEMA ..... 14
OBJETIVO GENERAL ..... 15
*Objetivos Específicos:* ..... *15*
ALCANCES Y LIMITACIONES ..... 15

**1. MARCO TEORICO** ..... **17**

1.1 BASES TEÓRICAS ..... 17
1.2 BASES CONCEPTUALES ..... 19

**2. DISEÑO METODOLÓGICO** ..... **22**

**3. RESULTADOS** ..... **31**

**4. DISCUSIÓN** ..... **33**

**5. CONCLUSIONES Y RECOMENDACIONES** ..... **35**

5.1 CONCLUSIONES ..... 35
5.2 RECOMENDACIONES ..... 36

**6. BIBLIOGRAFÍA** ..... ¡ERROR! MARCADOR NO DEFINIDO.

## Introducción

La seguridad informática es un tema crucial en la actualidad, especialmente para aquellas empresas que manejan datos sensibles y/o servicios críticos para su operación. Es por ello por lo que la Fundación OWASP (Open Web Application Security Project) constituye una comunidad de seguridad informática que trabaja para crear artículos, metodologías, documentación, herramientas y tecnologías que se liberan y pueden ser usadas por cualquiera. Dentro de estas herramientas y documentación ha identificado las 10 principales vulnerabilidades en aplicaciones web, conocidas como el TOP10 de OWASP. Estas vulnerabilidades pueden ser explotadas por ciberdelincuentes para acceder a información confidencial o causar daños al sistema.

En este contexto, la empresa Diebold Nixdorf dedicada al mantenimiento preventivo, correctivo, seguridad física y cibernética de cajeros automáticos de varias entidades financieras, cuenta con un aplicativo web denominado DocBase3 el cual recibe los incidentes de sus clientes corporativos del área financiera a nivel nacional e internacional, las cuales están sujetas a acuerdos de niveles de servicio (ANS) las monitorea y gestiona para dar solución a los problemas que estén presentando. Cuando estas ANS no se cumplen, generan una falta de disponibilidad del servicio por un ataque cibernético que podría traer consecuencias económicas importantes para la empresa.

Por lo anteriormente expuesto, en este trabajo se le propone a la empresa Diebold Nixdorf una consultoría que realice un análisis de vulnerabilidades para aplicativos webs con OWASP para identificar y remediar cualquier posible vulnerabilidad. De esta forma, se podrán tomar medidas preventivas para evitar situaciones de riesgo y garantizar la continuidad del servicio de manera segura y confiable.

## ANTECEDENTES Y JUSTIFICACION

Desde su fundación en 1859 Diebold Nixdorf ha mantenido un enfoque firme en tecnología, soluciones innovadoras y atención al cliente. Durante más de 160 años es líder mundial en comercio automatizado, digitaliza y transforma la forma en que las personas realizan operaciones bancarias.

Comprometido con la seguridad cibernética Diebold Nixdorf ha desarrollado una herramienta denominada DN (Vynamic security) la cual busca proporcionar soluciones de seguridad personalizada y eficaz para cada uno de sus clientes permitiendo proteger sus activos con tecnología de alta calidad

OWASP (Open Web Application Security Project) es una comunidad global que se dedica a mejorar la seguridad del software. Proporciona recursos, herramientas y directrices para ayudar a las organizaciones a desarrollar y mantener aplicaciones web seguras. Numerosas organizaciones y profesionales de todo el mundo han utilizado las metodologías y recursos de OWASP en sus proyectos de seguridad. Por lo cual mencionaremos algunos ejemplos de empresas y organizaciones conocidas que han utilizado las metodologías de OWASP:

Los gigantes de tecnología a nivel mundial como Google, Microsoft, IBM e incluso la fundación Mozilla ha utilizado varias metodologías de OWASP para mejorar la seguridad de sus productos y servicios. A continuación, se detallan algunas de las metodologías de OWASP que han adoptado:

- Se ha utilizado la lista OWASP Top 10 como referencia para identificar las diez principales vulnerabilidades en las aplicaciones web. Esta lista proporciona una visión general de las vulnerabilidades más críticas y comunes, como inyección SQL, cross-site scripting (XSS) y vulnerabilidades de autenticación. Microsoft ha utilizado esta lista como base para desarrollar controles y mitigaciones específicas en sus productos y servicios. **(Budzynski, Lepow, & Ovhal, 2023)**

- Proyecto OWASP AppSec Pipeline: La metodología del Proyecto OWASP AppSec Pipeline para desarrollar un enfoque de seguridad en el ciclo de vida del desarrollo de aplicaciones. Esta metodología establece un conjunto de pasos y actividades que deben realizarse en cada etapa del ciclo de vida, desde el diseño hasta la producción y el mantenimiento. Google ha utilizado esta metodología para integrar la seguridad en todas las fases del desarrollo de sus aplicaciones. **(Google Cloud, 2021)**

- Proyecto OWASP Web Security Testing Guide (WSTG): La guía de pruebas de seguridad de aplicaciones web de OWASP para evaluar la seguridad de sus aplicaciones. Esta guía proporciona una metodología completa para realizar pruebas de seguridad y evaluar vulnerabilidades en aplicaciones web. Google ha utilizado esta guía para definir sus propios procesos de prueba de seguridad y garantizar una cobertura exhaustiva en sus evaluaciones. **(Owasp, 2020)**

- Herramientas de OWASP: Varias herramientas desarrolladas por OWASP para mejorar la seguridad de sus aplicaciones. Por ejemplo, han utilizado OWASP ZAP (Zed Attack Proxy) para realizar pruebas de penetración y escaneo de vulnerabilidades en sus aplicaciones web. También han empleado otras herramientas de OWASP, como Dependency-Check, para identificar y abordar vulnerabilidades en las bibliotecas y componentes utilizados en sus aplicaciones.

En Colombia, varias organizaciones y profesionales de seguridad han utilizado las metodologías de OWASP para mejorar la seguridad de sus aplicaciones web y fortalecer sus procesos de desarrollo seguro. Mencionaremos algunos ejemplos de entidades en Colombia que han utilizado las metodologías de OWASP como entidades gubernamentales, Empresas de desarrollo de software, Instituciones académicas y comunidades de profesionales de seguridad informática quienes organizan eventos talleres y capacitaciones en los que comparten conocimientos sobre seguridad de aplicaciones web y como aplicar las metodologías de OWASP en sus entornos:

- Las entidades financieras como Bancolombia, Grupo AVAL y otras como entidades financieras líder en Colombia, tienen la responsabilidad de proteger la información financiera sensible de sus clientes y garantizar la seguridad de sus aplicaciones web y sistemas. En este sentido, es probable que hayan adoptado las metodologías de OWASP para abordar las vulnerabilidades comunes y mejorar la seguridad de sus aplicaciones.

- Grupo Éxito, una de las principales cadenas de supermercados en Colombia, ha implementado las metodologías de OWASP en el desarrollo de sus aplicaciones web y móviles. Han seguido las buenas prácticas de seguridad recomendadas por OWASP para mitigar las vulnerabilidades comunes y proteger los datos de sus clientes.

- Varias instituciones académicas en Colombia han utilizado las metodologías de OWASP como parte de sus programas de seguridad informática y desarrollo de software como La Universidad de los Andes, La Universidad Nacional, La Universidad de Antioquia.

**FORMULACION DEL PROBLEMA**

Diebold Nixdorf es una multinacional norteamericana que ofrece principalmente soluciones en la fabricación, venta, instalación y servicio de sistemas de transacciones de autoservicio cajeros automáticos y terminales postventa, productos de seguridad física y electrónica, entre otros servicios.

La empresa tiene un aplicativo web denominado DocBase3 el cual recibe los incidentes de los clientes corporaciones del área financiera a nivel nacional, para los cuales se tienen ANS (Acuerdos de Niveles de Servicio) que de no cumplirse conlleva a sanciones económicas.

Dado lo anterior es importante que se pruebe este aplicativo con las metodologías y herramientas de OWASP incluyendo el top 10 de vulnerabilidades identificadas por la Fundación OWASP, porque al presentarse un ataque cibernético que indisponga el servicio de recepción de los incidentes traería grandes consecuencias económicas para la empresa.

Por todo lo anteriormente expuesto, surge la pregunta: ¿es indispensable proponerle a la empresa Diebold Nixdorf una consultoría que realice un análisis de vulnerabilidades para aplicativos webs con la documentación, metodologías y herramientas de OWASP? Para que así se tomen las medidas preventivas correspondientes y no verse involucrado en sanciones económicas y de mala reputación a nivel empresa.

## OBJETIVO GENERAL

Proponer una Consultoría que realice un análisis de las vulnerabilidades del aplicativo web DocBase3 basado en metodologías de OWASP para la empresa Diebold Nixdorf.

***Objetivos Específicos:***

Describir un método para conocer el nivel de seguridad que tiene actualmente el aplicativo web DocBase3

Plantear la técnica de búsqueda y enumeración de vulnerabilidades de seguridad en el software web DocBase3 que garantice cubrir el mayor porcentaje de brechas de seguridad existentes.

Diseñar un análisis de cómo se van a comparar las vulnerabilidades de seguridad encontradas en el aplicativo web DocBase3 y las fallas enumeradas en el top 10 de OWASP, para así determinar los posibles métodos de solución disponibles.

## ALCANCES Y LIMITACIONES

En este documento se propone una consultoría a la empresa Diebold Nixdorf, dicha consultoría realiza un análisis de vulnerabilidades del aplicativo web DocBase3 perteneciente a la empresa, y se Aplica la metodología OWASP que posee alcance internacional. Es de aclarar, que la metodología OWASP es usada internacionalmente para identificar, evaluar y remediar vulnerabilidades existentes en aplicativa web, garantizando que se cumplan la triada CID[1] (confidencialidad, integridad y disponibilidad).

Esta propuesta abarca varios aspectos esenciales como los recursos financieros, ya que aplicar este análisis implican costos a la empresa, además, el recurso humano, debido a que se requiere del

[1] Triada CID: Es un modelo diseñado para guiar las políticas de seguridad de la información dentro de una organización [1].

personal clave de la organización, incluyendo el equipo de seguridad de la información y otros departamentos relevantes. También puede ser necesario contratar consultores externos para asistir en la implementación del SGSI (Sistema de gestión de seguridad de la información) para mitigar las vulnerabilidades que se encuentren.

Otro recurso que se tiene en cuenta en la consultoría son los recursos materiales, ya que implica la inversión en tecnología y herramientas para fortalecer la seguridad de los sistemas y aplicativos de la empresa. También puede requerir la actualización y mejora de las políticas y procedimientos de seguridad existentes.

En cuanto a las limitaciones de la propuesta se encuentra los términos de entrega y cumplimiento de especificaciones, ya que es importante tener en cuenta que la identificación y corrección de vulnerabilidades puede ser un proceso complejo y que puede llevar tiempo para garantizar que todas las vulnerabilidades se hayan abordado de manera adecuada.

Por supuesto, toda esta estrategia de implementación, en caso de realizarse, traerá consigo beneficios importantes, ya que la empresa Diebold Nixdorf tendrá más garantías para la seguridad de la información de sus clientes, evitando así pérdidas económicas por ataques informáticos, o incluso evitando multas por el mal uso o manejo de la información.

# 1. MARCO TEORICO

## 1.1 BASES TEÓRICAS

### Aplicativo Web DocBase3

Es una solución de software desarrollada por la empresa Diebold Nixdorf diseñada específicamente para el sector financiero. Permite gestionar, a través de módulos, los requerimientos de servicio. Proporciona un entorno eficiente para la captura, almacenamiento, administración, recuperación de documentos y datos relacionados con la generación, coordinación y seguimiento de incidentes de cada entidad financiera.

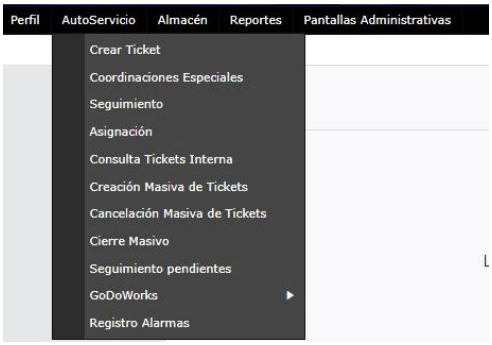



**Figura 1.** Nota. Módulos del aplicativo web DocBase3 para la gestión de incidentes (Copyright © Diebold Nixdorf, Incorporated, 2019)

| Ticket | Novedades | Luno | Fecha Efectiva | Fecha Generacion | Marca | Cliente | Sitio | Ciudad |
|---|---|---|---|---|---|---|---|---|
| 7023590 | | 4248 | 25/5/2023 09:30 | 08/3/2023 15:59 | DIEBOLD NIXDORF | BANCOLOMBIA | CAJERO SUC PAZ DE ARIPORO 1 - 2 | Paz de Ariporo |
| 7220369 | | 6079 | 25/5/2023 10:00 | 24/5/2023 22:03 | WINCOR NIXDORF | BANCOLOMBIA | CAJERO ESD TEXACO ACACIAS | Acacías |
| 7220413 | | 5543 | 25/5/2023 08:00 | 25/5/2023 03:21 | WINCOR NIXDORF | BANCOLOMBIA | CAJERO AVENIDA JUNIN 1 - 2 | Medellín |
| 7220477 | | 7319 | 25/5/2023 09:09 | 25/5/2023 07:09 | DIEBOLD | BANCOLOMBIA | CAJERO MF LA ESTACION IBAGUE 2 | Ibagué |
| 7220488 | | 2922 | 25/5/2023 11:00 | 25/5/2023 07:16 | WINCOR NIXDORF | SERVIBANCA | AERO_CTGENA II | Cartagena |
| 7220491 | | 8989 | 25/5/2023 15:17 | 25/5/2023 07:17 | DIEBOLD | BANCOLOMBIA | CAJERO SAN LUIS DE SINCE | Since |
| 7220492 | | 2171 | 25/5/2023 11:00 | 25/5/2023 07:17 | DIEBOLD | SERVIBANCA | HOSPITAL SAN CARLOS | Bogotá |
| 7219428 | | 1879 | 25/5/2023 08:00 | 24/5/2023 15:03 | DIEBOLD | BBVA COLOMBIA | Olimpica Buenaventura | Buenaventura |
| 7220388 | | 6383 | 25/5/2023 10:00 | 24/5/2023 23:33 | DIEBOLD | BANCOLOMBIA | CAJERO SUPERINTER JAMUNDI 1 - 2 | Jamundí |
| 7209605 | | 2653 | 25/5/2023 11:00 | 19/5/2023 16:58 | DIEBOLD | BANCO DE BOGOTA | OFICINA SAN VICENTE DE CAGUAN I - II | San Vicente del Caguan |
| 7215865 | | 6493 | 25/5/2023 09:00 | 23/5/2023 10:34 | WINCOR NIXDORF | BANCOLOMBIA | CAJERO SUC URIBIA 1 - 3 | Uribia |

**Figura 2.** Nota. Módulo de asignación de incidentes permite visualizar los tickets generados por las entidades bancarias, Muestra los tickets que están activos sin ingeniero asignado, junto con la información básica y los SLA pactados. (Copyright © Diebold Nixdorf, Incorporated, 2019)

**OWASP**

OWASP (Open Web Application Security Project) ofrece una variedad de metodologías y recursos para mejorar la seguridad de las aplicaciones web. A continuación, se presenta una descripción general de las principales metodologías de OWASP:

OWASP Top 10: La lista OWASP Top 10 es una recopilación de las diez vulnerabilidades más críticas que se encuentran comúnmente en las aplicaciones web. Esta lista se actualiza periódicamente para reflejar las amenazas emergentes. Las vulnerabilidades incluidas en el Top 10 pueden ser desde inyecciones de código hasta problemas de autenticación y autorización insegura. La lista sirve como una guía esencial para que los desarrolladores y profesionales de seguridad comprendan y aborden las principales amenazas en las aplicaciones web.

OWASP Testing Guide: La Guía de Pruebas de OWASP proporciona una metodología y un conjunto de técnicas para realizar pruebas de seguridad en aplicaciones web. Estas pruebas incluyen pruebas de penetración, pruebas de vulnerabilidad y pruebas de seguridad automatizadas. La guía ayuda a los profesionales de seguridad a evaluar las aplicaciones en busca de vulnerabilidades y puntos débiles, lo que les permite tomar medidas para corregirlos.

OWASP Secure Software Development Lifecycle (Secure SDLC): Esta metodología se centra en la integración de la seguridad en todas las etapas del ciclo de vida de desarrollo de software. Promueve la implementación de buenas prácticas de seguridad desde el diseño y la arquitectura hasta las pruebas y la implementación de las aplicaciones. La metodología Secure SDLC ayuda a garantizar que las aplicaciones se desarrollen de manera segura desde el principio y que se realicen evaluaciones y pruebas continuas para detectar y corregir vulnerabilidades.

OWASP Application Security Verification Standard (ASVS): El estándar ASVS de OWASP es una guía exhaustiva para verificar y evaluar la seguridad de las aplicaciones web. Proporciona una lista de requisitos y controles de seguridad que deben considerarse durante el desarrollo y la implementación de una aplicación. El ASVS abarca una amplia gama de áreas, desde la autenticación y la gestión de sesiones hasta la criptografía y la protección contra ataques de seguridad.

OWASP Code Review Guide: Esta guía proporciona un marco y una metodología para realizar revisiones de código en busca de vulnerabilidades de seguridad en las aplicaciones web. Ayuda a los revisores de código a identificar y corregir problemas de seguridad en el código fuente de las

aplicaciones, como inyecciones de código, vulnerabilidades de configuración y deficiencias en la autenticación y la autorización.

Estas son las principales metodologías de OWASP que se utilizan para mejorar la seguridad de las aplicaciones web. Cada una de ellas proporciona un enfoque específico para abordar diferentes aspectos de la seguridad, desde la identificación de vulnerabilidades hasta la integración de buenas prácticas en el desarrollo y la revisión del código. La combinación de estas metodologías puede ayudar a las organizaciones a construir aplicaciones más seguras y proteger la información confidencial de los usuarios.

## 1.2 BASES CONCEPTUALES

La metodología OWASP (Open Web Application Security Project) identifica y clasifica varias vulnerabilidades comunes en aplicaciones web. A continuación, se describen brevemente algunas de las vulnerabilidades más conocidas según la metodología OWASP:

**Inyección de código (Injection):** Esta vulnerabilidad ocurre cuando los datos no se validan o se filtran adecuadamente, lo que permite que los atacantes inserten código malicioso en las consultas SQL, comandos del sistema operativo u otros lenguajes de programación. Puede conducir a la ejecución no autorizada de comandos o al acceso no autorizado a datos sensibles.

**Cross-Site Scripting (XSS):** Esta vulnerabilidad permite que los atacantes inserten scripts maliciosos en páginas web visitadas por otros usuarios. Los ataques XSS pueden permitir que los atacantes roben información confidencial, como cookies de sesión, o redirijan a los usuarios a sitios web falsificados.

**Cross-Site Request Forgery (CSRF):** En este tipo de vulnerabilidad, los atacantes engañan a los usuarios para que realicen acciones no deseadas en un sitio web sin su conocimiento o consentimiento. Esto puede conducir a cambios no autorizados en la información, como modificar configuraciones o realizar acciones en nombre del usuario.

**Exposición de datos sensibles (Sensitive Data Exposure):** Esta vulnerabilidad ocurre cuando los datos sensibles, como contraseñas o información financiera, se almacenan o transmiten de manera insegura. Si los atacantes obtienen acceso a estos datos, podrían utilizarlos para actividades maliciosas, como el robo de identidad.

**Configuración incorrecta de seguridad (Security Misconfiguration):** Esta vulnerabilidad se produce cuando los sistemas o componentes de la aplicación web están configurados de manera incorrecta o insegura. Esto puede incluir permisos de archivo incorrectos, configuraciones de seguridad débiles o información de depuración habilitada, lo que facilita los ataques y la explotación de vulnerabilidades.

**Control de acceso insuficiente (Broken Access Control):** En esta vulnerabilidad, los mecanismos de control de acceso no se implementan correctamente, lo que permite que los usuarios no autorizados accedan a recursos o realicen acciones que deberían estar restringidas. Esto puede conducir a la divulgación de información confidencial o la modificación no autorizada de datos.

**Deserialización insegura (Insecure Deserialization):** Esta vulnerabilidad ocurre cuando los datos serializados no se validan adecuadamente durante el proceso de deserialización. Los atacantes pueden aprovechar esto para ejecutar código malicioso, realizar ataques de denegación de servicio o manipular datos.

Al aplicar las metodologías para mejorar la seguridad del aplicativo web de gestión de requerimientos e incidentes de servicio técnico DocBase3 se recomienda seguir los siguientes pasos basados en la guía de documentación de OWASP.

- **Identificación de requisitos de seguridad:** Comienza por identificar los requisitos específicos de seguridad para el aplicativo web de gestión de requerimientos e incidentes de servicio técnico. Esto implica determinar qué datos sensibles se manejarán, los roles y permisos de los usuarios, los requisitos de autenticación y autorización, y cualquier otra consideración relevante para la seguridad.

- **OWASP Top 10:** Familiarízate con la lista OWASP Top 10 y verifica que el aplicativo web no presente ninguna de estas vulnerabilidades comunes. Asegurarse de aplicar las contramedidas adecuadas para mitigar o eliminar las vulnerabilidades identificadas.

- **Diseño seguro:** Adoptar un enfoque de diseño seguro para el aplicativo web. Esto implica considerar aspectos como la autenticación fuerte, el manejo adecuado de sesiones, el cifrado de datos en reposo y en tránsito, la validación de entradas y salidas, y el uso de componentes y bibliotecas seguras.

- **Pruebas de seguridad:** Realizar pruebas de seguridad regulares utilizando la metodología y las técnicas de pruebas de OWASP. Esto incluye pruebas de penetración, pruebas de vulnerabilidad y pruebas de seguridad automatizadas. Identifica posibles puntos débiles y vulnerabilidades en tu aplicativo y tomar medidas para corregirlos.

- **Revisión de código:** Realizar revisiones de código en busca de posibles vulnerabilidades de seguridad. Utilizar la guía de revisión de código de OWASP para identificar problemas comunes, como inyecciones de código, validación insuficiente y deficiencias en la gestión de sesiones.

- **Capacitación y concientización:** Proporcionar capacitación y concientización en seguridad a los desarrolladores y al personal involucrado en el mantenimiento y la gestión del aplicativo. Asegurarse de que comprendan las mejores prácticas de seguridad de OWASP y sepan cómo implementarlas correctamente.

- **Monitorización y respuesta a incidentes:** Implementar mecanismos de monitorización continua para detectar y responder a posibles incidentes de seguridad. Establece alertas y registros de auditoría para identificar actividades sospechosas o anómalas. Define y practica un plan de respuesta a incidentes para abordar cualquier violación de seguridad de manera oportuna y eficiente.

- **Actualizaciones y parches:** Mantener el aplicativo web actualizado con las últimas correcciones de seguridad y parches. Realiza evaluaciones regulares de la seguridad para identificar nuevas amenazas y vulnerabilidades y asegúrate de aplicar las actualizaciones necesarias.

## 2. Diseño Metodológico

Para cumplir el objetivo de analizar el estado actual del aplicativo web DocBase3, se debe realizar el análisis de vulnerabilidades utilizando la metodología de OWASP (Open Web Application Security Project), para lo cual podemos seguir las siguientes indicaciones:

- **Planificación y alcance:**
  - Definir los objetivos del análisis de vulnerabilidades y los límites del alcance.
  - Identificar las aplicaciones web de Diebold Nixdorf que se someterán a la evaluación.
  - Establecer un cronograma para la realización del análisis.

- **Recopilación de información:**
  - Obtener información detallada sobre las aplicaciones web, como su arquitectura, tecnologías utilizadas, puntos de entrada, etc.
  - Realizar un inventario de los activos y recursos relevantes asociados con las aplicaciones.

- **Análisis estático del código:**
  - Utilizar herramientas de análisis estático de código para buscar vulnerabilidades conocidas en el código fuente de las aplicaciones.
  - Identificar posibles problemas de seguridad, como la falta de validación de entradas, vulnerabilidades de inyección, manejo inadecuado de errores, etc.

- **Análisis dinámico:**

- Realizar pruebas de penetración utilizando herramientas automatizadas o manuales para identificar vulnerabilidades en tiempo de ejecución.
- Evaluar las aplicaciones web en busca de vulnerabilidades comunes, como las inyecciones de SQL, la falta de autenticación y autorización adecuadas, las vulnerabilidades de cross-site scripting (XSS), etc.

- **Análisis de configuración del servidor:**
  - Verificar la configuración del servidor web y de la infraestructura subyacente para asegurar de que se sigan las mejores prácticas de seguridad.
  - Examinar la configuración de los firewalls, los certificados SSL/TLS, las políticas de acceso, etc.

- **Análisis de sesiones y gestión de identidad:**
  - Evaluar la forma en que se gestionan las sesiones de usuario y las operaciones de autenticación y autorización.
  - Buscar debilidades en la generación y gestión de tokens de sesión, la falta de cifrado en las comunicaciones, las contraseñas débiles, etc.

- **Informe de resultados:**
  - Documentar todas las vulnerabilidades encontradas, incluyendo detalles sobre su impacto y posibles medidas correctivas.
  - Proporcionar recomendaciones claras y específicas para abordar las vulnerabilidades identificadas.
  - Clasificar las vulnerabilidades según su gravedad y prioridad.

Realizando el anterior análisis inicial sobre el aplicativo web DocBase3, podremos obtener un estatus inicial de la aplicación.

Para cumplir el objetivo de buscar, enumerar y cubrir el mayor porcentaje de brechas de seguridad existentes, se debe realizar una implementación adecuada de la metodología OWASP, esto ayudara a garantizar que las aplicaciones sean seguras y protejan la información confidencial de los usuarios. Es importante tener en cuenta que la seguridad es un proceso continuo y que las aplicaciones deben

ser evaluadas regularmente para identificar nuevas vulnerabilidades y corregirlas antes de que puedan ser explotadas por los atacantes. se debe realizar una implementación adecuada de Priorización de vulnerabilidades**,** ya que es importante priorizar las vulnerabilidades identificadas según su gravedad, facilidad de explotación y otros criterios relevantes para la seguridad. Esto permitirá tomar decisiones informadas sobre qué vulnerabilidades deben ser corregidas primero y qué medidas de seguridad deben ser implementadas para proteger el sistema o aplicación.

OWASP proporciona una metodología para priorizar las vulnerabilidades, la cual consta de cinco pasos:

- **Identificación de vulnerabilidades:** En este primer paso se deben identificar todas las posibles vulnerabilidades presentes en el sistema o aplicación. Esto se puede lograr a través de diversas técnicas, como el escaneo de vulnerabilidades, pruebas de penetración, revisión de código, entre otras.

- **Evaluación de la gravedad de la vulnerabilidad:** Una vez identificadas las vulnerabilidades, se debe evaluar su gravedad. Para esto, OWASP recomienda utilizar una escala de gravedad, que puede ser numérica (por ejemplo, del 1 al 10) o alfanumérica (por ejemplo, crítica, alta, media, baja). La gravedad de una vulnerabilidad se puede determinar considerando factores como el impacto en la confidencialidad, integridad y disponibilidad del sistema o datos, la facilidad de explotación, la cantidad de usuarios afectados, entre otros.

- **Evaluación de la facilidad de explotación:** Además de la gravedad de la vulnerabilidad, se debe evaluar la facilidad con la que un atacante podría explotarla. Esta evaluación se puede basar en factores como la complejidad de la explotación, la disponibilidad de herramientas de ataque, la cantidad de conocimiento técnico que se requiere para explotar la vulnerabilidad, entre otros. Las vulnerabilidades que son fáciles de explotar deben recibir una prioridad más alta que las que son más difíciles de explotar.

- **Evaluación de la relevancia de la vulnerabilidad:** También es importante evaluar la relevancia de la vulnerabilidad para el sistema o aplicación en cuestión. Por ejemplo, una vulnerabilidad que afecta a un componente crítico del sistema debe recibir una prioridad más alta que una que afecta a un componente menos importante. Para realizar esta evaluación,

se deben considerar factores como el valor de los datos que maneja la aplicación, el nivel de acceso que tienen los usuarios a la aplicación, entre otros.

- **Evaluación de la probabilidad de explotación:** Por último, se debe evaluar la probabilidad de que la vulnerabilidad sea explotada. Esta evaluación se basa en factores como la frecuencia de exposición de la vulnerabilidad, la frecuencia de ataques a la aplicación, la disponibilidad de herramientas de ataque, entre otros. Las vulnerabilidades que tienen una alta probabilidad de ser explotadas deben recibir una prioridad más alta.

La priorización de vulnerabilidades debe basarse en una combinación de estos factores, dependiendo del contexto específico del sistema o aplicación en cuestión. Es importante tener en cuenta que la priorización de vulnerabilidades es un proceso continuo y que debe realizarse de manera regular, ya que nuevas vulnerabilidades pueden surgir con el tiempo.

Para cumplir con el objetivo de comparar las vulnerabilidades de seguridad encontradas con el top 10 de OWASP se deben seguir los siguientes pasos:

- **Identificar las vulnerabilidades:** Para comenzar, es importante identificar todas las vulnerabilidades presentes en el sistema o aplicación que se está evaluando. Estas pueden ser descubiertas a través de un escaneo de vulnerabilidades, una auditoría de seguridad o mediante la revisión del código fuente.

- **Clasificar las vulnerabilidades:** Una vez identificadas las vulnerabilidades, es necesario clasificarlas según el tipo de ataque que pueden permitir, su nivel de gravedad, la facilidad de explotación y otros criterios relevantes para la seguridad.

- **Comparar con el top 10 de OWASP:** A continuación, se debe comparar cada una de las vulnerabilidades identificadas con los diez riesgos de seguridad más críticos según OWASP. Estos son los siguientes:
    - Inyección de código SQL (SQL Injection)
    - Lectura/escritura no autorizada en archivos (Broken Authentication and Session Management)
    - Ataques a través de sitios web no seguros (Cross-Site Scripting - XSS)

- Robo de identidad (Broken Access Control)
- Falsificación de petición en sitios cruzados (Cross-Site Request Forgery - CSRF)
- Problemas de seguridad en configuraciones (Security Misconfiguration)
- Uso de componentes con vulnerabilidades conocidas (Insecure Components)
- Redirecciones y reenvíos no válidos (Unvalidated Redirects and Forwards)
- Problemas de autenticación y autorización en la capa de negocio (Business Logic and Data Validation)
- Exposición de datos sensibles (Sensitive Data Exposure)

Para cada vulnerabilidad identificada, se debe evaluar si es comparable a alguna de las vulnerabilidades del top 10 de OWASP, y de ser así, cómo se compara en términos de gravedad, facilidad de explotación y otros factores relevantes.
Por supuesto, a continuación, se profundizará en cada punto del top 10 de OWASP y se explicará su metodología:

- Inyección de código SQL (SQL Injection): La inyección de código SQL ocurre cuando un atacante envía datos maliciosos a una aplicación web que utiliza una base de datos. Esta vulnerabilidad permite al atacante ejecutar comandos SQL no autorizados que pueden acceder, modificar o eliminar información de la base de datos.

  La metodología de OWASP para prevenir esta vulnerabilidad incluye:

  - Validación de entrada: La entrada del usuario debe ser validada y filtrada para evitar la inyección de código malicioso.
  - Parámetros de consulta parametrizados: Los parámetros de consulta deben ser parametrizados para evitar la inyección de código SQL.
  - Evitar la concatenación de consultas: No se deben concatenar consultas SQL en una cadena de texto que contenga entradas de usuario.

- Lectura/escritura no autorizada en archivos (Broken Authentication and Session Management): La gestión incorrecta de la autenticación y la gestión de sesiones puede permitir a un atacante obtener acceso no autorizado a información confidencial, como nombres de usuario, contraseñas y otros datos privados.

  La metodología de OWASP para prevenir esta vulnerabilidad incluye:

- Autenticación y autorización sólidas: Las contraseñas deben ser seguras, la autenticación de dos factores debe ser utilizada y la autorización debe ser implementada correctamente.
- Gestionar sesiones de manera segura: Las sesiones deben ser controladas y finalizadas correctamente cuando el usuario sale o la sesión ha expirado.

- Ataques a través de sitios web no seguros (Cross-Site Scripting - XSS): Los ataques XSS ocurren cuando un atacante envía un script malicioso a un sitio web que se ejecuta en el navegador web de la víctima. Esto permite al atacante robar información confidencial del usuario, como contraseñas, cookies, historial del navegador y otras credenciales.

  La metodología de OWASP para prevenir esta vulnerabilidad incluye:
  - Validación y filtrado de entrada: Los datos de entrada deben ser validados y filtrados para evitar la ejecución de código malicioso.
  - Evitar la concatenación de código HTML: No se deben concatenar cadenas de HTML en una cadena de texto que contenga entradas de usuario.
  - Usar un marco de trabajo de seguridad: Los marcos de trabajo de seguridad, como Content Security Policy (CSP), pueden prevenir ataques XSS.

- Robo de identidad (Broken Access Control): La falta de controles de acceso adecuados puede permitir que un atacante obtenga acceso no autorizado a información confidencial o realice acciones maliciosas en el sistema.

  La metodología de OWASP para prevenir esta vulnerabilidad incluye:
  - Autenticación y autorización sólidas: Las contraseñas deben ser seguras, la autenticación de dos factores debe ser utilizada y la autorización debe ser implementada correctamente.
  - Control de acceso basado en roles: Los usuarios deben tener permisos limitados y solo pueden acceder a los recursos a los que se les ha dado acceso explícitamente.
  - Control de acceso a nivel de objeto: Los usuarios deben tener permisos limitados para acceder a los objetos dentro del sistema.

- Falsificación de petición en sitios cruzados (Cross-Site Request Forgery - CSRF): Los ataques CSRF ocurren cuando un atacante envía una solicitud no autorizada a un sitio web

que se ejecuta en el navegador web de la víctima. Esto permite al atacante realizar acciones maliciosas en nombre de la víctima, como cambiar la contraseña o realizar una transferencia de dinero.

La metodología de OWASP para prevenir esta vulnerabilidad incluye:

- Token de seguridad: Se debe implementar un token de seguridad para evitar que los ataques CSRF se aprovechen. El token debe ser único para cada sesión y debe ser incluido en cada solicitud que el usuario realice.
- Verificación del origen de la solicitud: Se debe verificar el origen de la solicitud para asegurarse de que proviene del sitio web legítimo y no de un sitio malicioso.
- Validación de entrada: Los datos de entrada deben ser validados y filtrados para evitar que los atacantes envíen solicitudes maliciosas.

- Vulnerabilidades en componentes externos (Security Misconfiguration): Las configuraciones incorrectas del servidor, del software o de los componentes externos pueden permitir a los atacantes acceder a información confidencial o realizar acciones maliciosas en el sistema.

  La metodología de OWASP para prevenir esta vulnerabilidad incluye:

  - Configuración segura del servidor: El servidor debe estar configurado de manera segura para evitar que los atacantes puedan aprovechar las vulnerabilidades de configuración.
  - Actualización de software y componentes: El software y los componentes deben actualizarse regularmente para corregir las vulnerabilidades conocidas.
  - Restricción de acceso a recursos sensibles: Los recursos sensibles, como los archivos de configuración y las claves de cifrado, deben estar restringidos y protegidos.

- Uso inseguro de funciones criptográficas (Insecure Cryptographic Storage): El almacenamiento inseguro de información confidencial, como contraseñas y tarjetas de crédito, puede permitir a los atacantes acceder a esta información y utilizarla para fines maliciosos.

La metodología de OWASP para prevenir esta vulnerabilidad incluye:

- Almacenamiento seguro de contraseñas: Las contraseñas deben almacenarse de manera segura utilizando técnicas de cifrado robustas, como hash y sal.
- Protección de datos sensibles: Los datos sensibles, como las tarjetas de crédito, deben estar protegidos utilizando técnicas de cifrado robustas.
- Configuración segura de la infraestructura: La infraestructura debe estar configurada de manera segura para evitar que los atacantes puedan acceder a la información confidencial.
-

- Insuficiente autenticación y autorización (Insufficient Authentication and Authorization): La autenticación y la autorización incorrectas o insuficientes pueden permitir que los atacantes accedan a información confidencial o realicen acciones maliciosas en el sistema.

  La metodología de OWASP para prevenir esta vulnerabilidad incluye:

  - Autenticación y autorización sólidas: Las contraseñas deben ser seguras, la autenticación de dos factores debe ser utilizada y la autorización debe ser implementada correctamente.
  - Control de acceso basado en roles: Los usuarios deben tener permisos limitados y solo pueden acceder a los recursos a los que se les ha dado acceso explícitamente.

- Falta de control de acceso a nivel de negocio (Inadequate Security Configurations): Las políticas de control de acceso inadecuadas pueden permitir que los atacantes accedan a información confidencial o realicen acciones maliciosas en el sistema.

  La metodología de OWASP para prevenir esta vulnerabilidad incluye:

  - Políticas de control de acceso sólidas: Las políticas de control de acceso deben ser implementadas y aplicadas correctamente para garantizar que los usuarios solo tengan acceso a los recursos que necesitan.
  - Separación de roles y privilegios: Los roles y los privilegios deben estar separados para garantizar que los usuarios solo tengan acceso a los recursos que necesitan para realizar sus funciones.

- Protección de datos confidenciales: Los datos confidenciales deben estar protegidos y solo deben ser accesibles para aquellos usuarios que necesitan acceder a ellos.

- Configuraciones de seguridad insuficientes (Insufficient Security Configurations): La configuración inadecuada del software y los componentes puede permitir que los atacantes accedan a información confidencial o realicen acciones maliciosas en el sistema.

La metodología de OWASP para prevenir esta vulnerabilidad incluye:

- Configuración segura de servidores y componentes: Los servidores y componentes deben estar configurados de manera segura y actualizados regularmente para corregir las vulnerabilidades conocidas.
- Restricción de acceso a recursos sensibles: Los recursos sensibles, como los archivos de configuración y las claves de cifrado, deben estar restringidos y protegidos.
- Verificación de configuraciones de seguridad: Las configuraciones de seguridad deben ser verificadas regularmente para detectar posibles vulnerabilidades y corregirlas antes de que puedan ser explotadas por los atacantes. (Owasp, 2020)

## 3. Resultados

Al aplicar un análisis de vulnerabilidades utilizando la metodología OWASP en el aplicativo web de gestión de incidentes y solicitud de soporte técnico DocBase3, se pueden obtener los siguientes resultados:

- Identificación de vulnerabilidades: El análisis revelará las vulnerabilidades presentes en el aplicativo web, como inyección de código, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), exposición de datos sensibles, configuraciones incorrectas de seguridad, control de acceso insuficiente, deserialización insegura, entre otras. Cada vulnerabilidad se clasifica y se proporcionará una descripción detallada de su impacto potencial en la seguridad del aplicativo.

- Priorización de riesgos: Basándose en la metodología OWASP, se evaluará la gravedad de cada vulnerabilidad identificada. Se asignarán niveles de riesgo para determinar qué vulnerabilidades deben abordarse primero en función de su impacto potencial y la probabilidad de que sean explotadas por un atacante.

- Recomendaciones de mitigación: Para cada vulnerabilidad identificada, se proporcionan recomendaciones claras y detalladas sobre cómo mitigar los riesgos. Estas recomendaciones pueden incluir cambios en el código fuente, actualizaciones de bibliotecas o frameworks, configuraciones de seguridad, validación de entrada de datos, autenticación y autorización mejoradas, entre otros. Las recomendaciones estarán alineadas con las mejores prácticas de seguridad de OWASP y se adaptarán a las características específicas del aplicativo web.

- Comparación con el Top 10 de OWASP: Al comparar los resultados del análisis con el Top 10 de OWASP, se evaluará si el aplicativo web tiene las vulnerabilidades más críticas y comunes que se encuentran en esa lista. Esto ayudará a determinar si el sistema está expuesto a los riesgos más significativos y si se están aplicando las mejores prácticas de seguridad recomendadas por OWASP.

- Plan de acción: Se elaborará un plan de acción detallado que indique los pasos necesarios para abordar las vulnerabilidades y mejorar la seguridad del aplicativo web. El plan de acción incluirá una secuencia lógica de tareas, asignación de responsabilidades y plazos estimados para cada actividad. Esto permitirá al equipo encargado de la seguridad del aplicativo implementar las mejoras de manera efectiva y oportuna.

- Informe final: Se entregará un informe final completo que resuma los resultados del análisis de vulnerabilidades. Este informe contendrá una descripción detallada de cada vulnerabilidad identificada, su impacto potencial, el nivel de riesgo asignado, las recomendaciones de mitigación y el plan de acción propuesto. El informe también puede incluir gráficos, tablas y evidencia técnica para respaldar las conclusiones y recomendaciones.

## 4. Discusión

Utilizando la metodología OWASP para determinar las vulnerabilidades en la aplicación DocBase3 puede arrojar resultados significativos. A través de la identificación, evaluación y priorización de las vulnerabilidades, se pueden obtener una serie de hallazgos importantes que requieren atención y acción.

En primer lugar, se pueden identificar diversas vulnerabilidades en el sistema. Estas vulnerabilidades representan puntos débiles en la seguridad de la aplicación y podrían ser aprovechadas por atacantes para acceder a información confidencial, realizar acciones maliciosas o comprometer la integridad del sistema. La identificación precisa de estas vulnerabilidades proporciona una visión clara de los riesgos y desafíos que se deben abordar.

La evaluación de la gravedad de las vulnerabilidades puede permitir establecer una jerarquía de riesgos, clasificando las vulnerabilidades según su impacto potencial. Esto es crucial para priorizar los esfuerzos y recursos de seguridad en áreas críticas. Al asignar una escala de gravedad, se pueden enfocar los recursos en la corrección de las vulnerabilidades más graves y urgentes, evitando una distribución desigual de los recursos de seguridad.

La evaluación de la facilidad de explotación y la probabilidad de explotación también son aspectos clave del análisis. Estos factores permitirán comprender la viabilidad de un ataque y la posibilidad real de que las vulnerabilidades sean aprovechadas. Al considerar estos aspectos, se pueden tomar decisiones informadas sobre qué vulnerabilidades abordar primero, brindando una respuesta proactiva y enfocada a los riesgos más inmediatos y probables.

La relevancia de las vulnerabilidades para el sistema o aplicación es otro factor importante que se ha tenido en cuenta. Esta evaluación ayudara a comprender cómo las vulnerabilidades podrían afectar directamente el funcionamiento y la seguridad del sistema. Al considerar la relevancia, se puede adaptar la estrategia de seguridad para abordar los aspectos más críticos y proteger los activos más valiosos.

El análisis para determinar las vulnerabilidades con la metodología OWASP brinda una visión integral de los riesgos de seguridad en un sistema o aplicación. Los resultados obtenidos a través de la identificación, evaluación y priorización de las vulnerabilidades permiten tomar decisiones

informadas sobre las medidas de seguridad necesarias. Es fundamental abordar las vulnerabilidades de manera proactiva y continua, implementando las medidas adecuadas para proteger la integridad y confidencialidad de la información, así como para garantizar el correcto funcionamiento del sistema. El análisis proporciona una base sólida para fortalecer la seguridad y reducir la exposición a posibles ataques.

# 5. Conclusiones y recomendaciones

## 5.1 Conclusiones

Tras implementar un análisis para la seguridad de un aplicativo web DocBase3 basado en OWASP, se pueden extraer las siguientes conclusiones:

- **Mayor conciencia de las vulnerabilidades comunes:** Se obtiene una comprensión más profunda de las vulnerabilidades comunes que pueden existir en el aplicativo web. Esto permite tomar medidas proactivas para abordarlas y reducir el riesgo de exposición a amenazas.

- **Identificación y mitigación de vulnerabilidades:** Se ayudo a describir un método para identificar las vulnerabilidades específicas presentes en el aplicativo web. Estas vulnerabilidades pueden ser abordadas mediante la implementación de contramedidas y mejores prácticas de seguridad recomendadas por OWASP, lo que fortalece la postura de seguridad del aplicativo.

- **Mejora de la seguridad en todas las etapas del ciclo de vida del desarrollo:** Se integra la seguridad en todas las etapas del ciclo de vida del desarrollo del aplicativo web. Esto implica considerar la seguridad desde el diseño y la implementación hasta las pruebas y el mantenimiento continuo. Como resultado, se establece una base sólida de seguridad desde el inicio del desarrollo.

- **Aumento de la protección de datos sensibles:** Se identifico como se deben implementar controles y medidas para proteger los datos sensibles manejados por el aplicativo web. Esto incluye el cifrado adecuado de los datos, la validación de entradas y salidas, y la implementación de prácticas seguras de autenticación y autorización. Como resultado, se fortalece la confidencialidad y la integridad de los datos.

- **Mayor confianza y satisfacción del usuario:** Se demuestra un compromiso con la seguridad del aplicativo web y la protección de los datos de los usuarios. Esto genera mayor confianza y satisfacción entre los usuarios, lo que puede impulsar la reputación y el éxito del aplicativo.

- **Preparación para amenazas emergentes:** OWASP se mantiene actualizado con las últimas amenazas y vulnerabilidades emergentes en el ámbito de la seguridad de aplicaciones web. Al realizar un análisis basado en OWASP, se está preparado para enfrentar las amenazas más recientes y se puede implementar medidas de mitigación adecuadas.

La implementación del análisis para la seguridad del aplicativo web DocBase3 basado en OWASP brinda una mayor seguridad, protección de datos y confianza en la aplicación. Ayuda a identificar y abordar vulnerabilidades comunes, mejorando así la postura de seguridad global y reduciendo el riesgo de violaciones de seguridad.

## 5.2 Recomendaciones

Después de implementar un análisis para la seguridad del aplicativo DocBase3 web basado en OWASP, se pueden hacer las siguientes recomendaciones:

- **Mantener actualizadas las mejores prácticas de seguridad:** Dado que las amenazas y las vulnerabilidades evolucionan constantemente, es importante mantenerse actualizado con las últimas mejores prácticas de seguridad de OWASP. Mantén un monitoreo regular de los recursos y las actualizaciones proporcionadas por OWASP para estar al tanto de las últimas recomendaciones de seguridad.

- **Realizar pruebas de seguridad periódicas:** No se trata de realizar un análisis de seguridad solo una vez, sino de establecer un enfoque continuo de pruebas de seguridad. Realiza pruebas periódicas utilizando las metodologías y herramientas de OWASP para identificar nuevas vulnerabilidades y evaluar la efectividad de las medidas de seguridad implementadas.

- **Implementar correcciones y mejoras:** Utiliza los resultados del análisis de seguridad basado en OWASP para implementar las correcciones y mejoras necesarias en el aplicativo web. Prioriza las vulnerabilidades encontradas y establece un plan de acción para solucionarlas. Asegúrate de asignar los recursos adecuados y establecer plazos realistas para abordar las vulnerabilidades identificadas.

- **Fomentar la conciencia y la capacitación en seguridad:** No solo es importante que el equipo de desarrollo esté al tanto de las mejores prácticas de seguridad, sino que todo el personal involucrado en el aplicativo web también esté consciente de los riesgos y sepa cómo contribuir a la seguridad. Proporciona capacitación y concienciación en seguridad a todo el personal, incluyendo a los desarrolladores, el personal de soporte y los usuarios finales.

- **Implementar un programa de respuesta a incidentes:** Establece un programa de respuesta a incidentes que permita abordar rápidamente cualquier violación de seguridad o incidente relacionado. Define los pasos a seguir, los responsables y los procedimientos para contener, investigar y mitigar los incidentes de seguridad. Practica y evalúa regularmente el programa para garantizar su efectividad.

- **Realizar evaluaciones de terceros:** Si el aplicativo web interactúa con proveedores externos o servicios de terceros, es recomendable realizar evaluaciones de seguridad en esos componentes externos. Asegúrate de que los proveedores cumplan con los estándares de seguridad adecuados y que se tomen las medidas necesarias para proteger los datos y la integridad del aplicativo.

- **Monitorear y auditar la seguridad:** Implementa un sistema de monitoreo y auditoría de seguridad para detectar actividades sospechosas y garantizar el cumplimiento de las políticas de seguridad. Utiliza herramientas de registro y monitoreo de eventos para identificar posibles amenazas y realizar un seguimiento de las acciones de los usuarios.

- **Promover una cultura de seguridad:** La seguridad no es solo responsabilidad del equipo de seguridad o del personal de TI, sino que debe ser un esfuerzo conjunto de toda la organización. Fomenta una cultura de seguridad en la que se promueva la conciencia, la responsabilidad y la colaboración en materia de seguridad en todos los niveles.

Implementar un análisis para la seguridad de un aplicativo web basado en OWASP es solo el primer paso. La seguridad debe ser un proceso continuo y en evolución para mantenerse al día con las amenazas cambiantes. Al seguir estas recomendaciones, podrás mejorar la seguridad de tu aplicativo web y reducir los riesgos asociados a las vulnerabilidades de seguridad.

## 6. Bibliografía


Budzynski, M., Lepow, D., & Ovhal, P. (22 de 04 de 2023). *Recomendaciones para mitigar las 10 principales amenazas de seguridad de las API de OWASP mediante API Management*. Obtenido de learn.microsoft: https://learn.microsoft.com/es-es/azure/api-management/mitigate-owasp-api-threats

Copyright © Diebold Nixdorf, Incorporated. (01 de 01 de 2019). *Manual de usario Docbase3*. Obtenido de dieboldnixdorf: dieboldnixdorf.sharepoint.com

Google Cloud. (12 de 12 de 2021). *Opciones de mitigación OWASP Top 10 2021 en Google Cloud*. Obtenido de Cloud Architecture Center: https://cloud.google.com/architecture/owasp-top-ten-mitigation?hl=es-419

Imagine Easy Solutions. (09 de 2016). *Normas APA*. Obtenido de http://normasapa.net/

Mendeley. (09 de 2016). *Mendeley*. Obtenido de https://www.mendeley.com/

Owasp. (03 de 12 de 2020). *Guía de pruebas de seguridad web de OWASP*. Obtenido de Owasp version 4.2: https://owasp.org/www-project-web-security-testing-guide/

Roy Rosenzweig Center for History and New Media. (09 de 2016). *Zotero*. Obtenido de https://www.zotero.org

Sáenz, J. (09 de 09 de 2016). *Wiki Sistemas Digitales*. Obtenido de http://sistdig.wikidot.com

Suárez, R. (2015). Energías renovables, Impacto y Efectividad. *Revista Científica Ingeniería y Desarrollo en Energía*, 48-53.

Thomson Reuters. (09 de 2016). *EndNote*. Obtenido de http://endnote.com/