Universidad Mariano Gálvez de Guatemala

Facultad de Ingeniería

Sede Chiquimulilla.

Catedrático:

Carmelo Estuardo Monterroso Mayen.

Curso:

Aseguramiento de la Calidad de Software.

Alumno:

Kevin José Santos Hernández

Carné:

1790-19-19560

Semestre:

10.

Chiquimulilla, 01 de agosto de 2025.

Guía de Mitigación de Vulnerabilidades OWASP Top 10 – 2021

Primera Parte: Guía OWASP Top 10 – 2021

El objetivo de esta guía es explicar, de manera clara y con lenguaje sencillo, las diez principales vulnerabilidades de seguridad en aplicaciones web, según la lista OWASP Top 10 del año 2021, y cómo prevenirlas.

1. Control de Acceso Roto (Broken Access Control)

¿Qué es?

Ocurre cuando los usuarios pueden acceder a funciones o datos que no deberían estar disponibles para ellos.

¿Cómo prevenirlo?

Asegurarse de que cada usuario solo tenga acceso a lo que le corresponde.

No ocultar funciones en la interfaz como método de seguridad.

Revisar que no se pueda acceder a funciones cambiando la URL o datos de entrada.

2. Fallos Criptográficos (Cryptographic Failures)

¿Qué es?

Cuando los datos sensibles como contraseñas o tarjetas de crédito no están bien protegidos.

¿Cómo prevenirlo?

Usar conexiones seguras (HTTPS).

Guardar contraseñas de forma cifrada.

No enviar datos importantes por correo o en texto simple.

3. Inyecciones (Injection)

¿Qué es?

Cuando un atacante pone comandos peligrosos en formularios o campos para controlar el sistema.

¿Cómo prevenirlo?

Validar y revisar la información que ingresan los usuarios.

No permitir que se ingresen símbolos especiales sin control.

Usar herramientas que limpien los datos automáticamente.

4. Diseño Inseguro (Insecure Design)

¿Qué es?

Cuando desde el inicio del desarrollo del sistema no se piensa en la seguridad.

¿Cómo prevenirlo?

Planear la seguridad desde el diseño.

Imaginar posibles ataques y cómo evitarlos.

Hacer pruebas desde las primeras etapas del sistema.

5. Configuración Incorrecta de Seguridad (Security Misconfiguration)

¿Qué es?

Cuando el sistema tiene configuraciones débiles o por defecto que permiten ataques.

¿Cómo prevenirlo?

Cambiar contraseñas por defecto.

Mantener el sistema actualizado.

Revisar y ajustar las configuraciones de seguridad.

6. Componentes Vulnerables y Desactualizados

¿Qué es?

Cuando se usan partes del sistema que están viejas o tienen fallas conocidas.

¿Cómo prevenirlo?

Mantener actualizados todos los programas y herramientas.

Verificar antes de instalar cualquier componente.

Eliminar lo que no se usa.

7. Fallos en la Identificación y Autenticación

¿Qué es?

Cuando el sistema permite entrar a usuarios sin verificar bien su identidad.

¿Cómo prevenirlo?

Usar contraseñas fuertes.

Pedir doble verificación (por ejemplo, código por SMS).

Cerrar sesiones después de cierto tiempo.

8. Fallos en la Integridad de Software y Datos

¿Qué es?

Cuando alguien puede cambiar el software o los datos sin que el sistema lo note.

¿Cómo prevenirlo?

Usar sistemas para verificar que los archivos no se han modificado.

No permitir que cualquier persona suba archivos al sistema.

Revisar qué se instala y de dónde proviene.

9. Fallo en la Monitorización y Registro de Seguridad

¿Qué es?

Cuando el sistema no detecta ni guarda información sobre ataques o errores.

¿Cómo prevenirlo?

Activar alertas para actividades sospechosas.

Guardar registros de los accesos y acciones de los usuarios.

Revisar frecuentemente esos registros.

10. Falsificación de Solicitudes del Lado del Servidor (SSRF)

¿Qué es?

Ocurre cuando un atacante engaña al servidor para que acceda a sitios internos que no deberían estar disponibles.

¿Cómo prevenirlo?

Revisar los enlaces o direcciones que los usuarios pueden ingresar.

No permitir que el sistema acceda a su propia red interna sin autorización.

Usar filtros que bloqueen direcciones sospechosas.

Glosario

- **Actualización**: Mejoras o arreglos que se hacen a los programas para corregir errores o agregar funciones nuevas.
- **Autenticación**: Comprobación que hace un sistema para confirmar que un usuario es quien dice ser.
- **Cifrado**: Técnica que transforma la información para que solo personas autorizadas puedan entenderla.

- **Contraseña fuerte:** Clave difícil de adivinar, que combina letras mayúsculas y minúsculas, números y símbolos.
- **HTTPS**: Protocolo seguro para acceder a páginas web. Protege la información que se envía entre el usuario y el sitio.
- **Servidor**: Computadora que almacena y entrega páginas web o datos a otras computadoras.
- **Verificación en dos pasos:** Método de seguridad que solicita dos formas de identificación, como una contraseña y un código enviado al celular.

Segunda Parte: Planes y Casos de Prueba

¿Qué son los Planes de Prueba?

Un plan de prueba es un documento que sirve como guía para verificar que un sistema o aplicación funcione correctamente. Este documento describe qué se va a probar, cómo se va a hacer, quién lo hará, con qué herramientas y en qué momento.

Se utiliza para organizar y planificar todo el proceso de pruebas antes de poner un sistema en funcionamiento. Es decir, el plan de prueba es como una hoja de ruta para asegurar la calidad del producto antes de que lo usen los usuarios.

Elementos comunes de un plan de prueba:

Objetivo de la prueba: qué se quiere comprobar.

Alcance: qué partes del sistema serán probadas.

Recursos necesarios: quién realizará la prueba y qué herramientas se usarán.

Cronograma: fechas de inicio y fin.

Criterios de éxito: cómo saber si el resultado es correcto.

Riesgos: posibles problemas durante las pruebas.

¿Para qué sirve un plan de prueba?

Detectar errores antes de que el sistema sea usado por los clientes.

Asegurar que el producto cumpla con lo que se espera.

Ahorrar tiempo y dinero evitando fallos futuros.

Establecer orden y responsabilidad en el proceso de validación.

¿Qué son los Casos de Prueba?

Los casos de prueba son situaciones específicas que se utilizan para comprobar que cada parte del sistema hace lo que debería hacer.

Cada caso de prueba incluye una entrada (por ejemplo, llenar un formulario) y un resultado esperado (por ejemplo, que se muestre un mensaje de confirmación). Se utilizan para verificar si una funcionalidad funciona bien o si hay algún error que corregir.

Un caso de prueba suele incluir:

ID del caso (por ejemplo, CP01).

Descripción de lo que se va a probar.

Datos de entrada que el usuario ingresará.

Pasos a seguir para hacer la prueba.

Resultado esperado.

Resultado obtenido (cuando se ejecuta).

Estado: si pasó o falló la prueba.

¿Para qué sirven los casos de prueba?

Comprobar cada funcionalidad del sistema.

Encontrar errores en situaciones específicas.

Garantizar que los cambios en el sistema no afecten otras funciones.

Documentar lo que ya fue probado y lo que falta por probar.

Ejemplo de Plan de Pruebas: Catálogo de Registros

Nombre del Proyecto: Sistema de Catálogo de Productos
Objetivo de la Prueba: Verificar que el catálogo permita crear, editar y eliminar registros correctamente.
Responsable: Área de Calidad

Fecha de inicio: 1 de agosto de 2025

Fecha de finalización: 3 de agosto de 2025

Herramientas utilizadas: Navegador web, hoja de cálculo para registro de resultados

Funciones a probar:

Agregar nuevo producto

Editar producto existente

Eliminar producto

Visualizar lista de productos

Casos de Prueba

ID	Caso de Prueba	Entrada	Resultado Esperado	Resultado Obtenido	Estado
CP01	Crear producto	Nombre: “Producto 1”	Producto guardado y visible en lista	Producto guardado correctamente	Aprobado
CP02	Editar producto	Cambiar nombre a “Producto 1 Actualizado”	Cambio guardado correctamente	Nombre actualizado	Aprobado
CP03	Eliminar producto	Eliminar “Producto 1 Actualizado”	Producto eliminado de la lista	Producto eliminado	Aprobado
CP04	Crear producto sin nombre	Campo vacío	Mostrar mensaje de error	Se muestra error: “Nombre requerido”	Aprobado