Proyecto 1: Cybersecurity Consulting

TrustShield Financial

Autores: Pablo González · Carlos Alcina · David Jiménez · Luis Carlos Romero

Área: Aplicaciones Web

Cliente: TrustShield Financial

Introducción

La ciberseguridad no es un gasto, sino una **inversión estratégica**.

El proyecto se centra en **detectar vulnerabilidades críticas** en las aplicaciones web del cliente TrustShield Financial.

Metodología en dos fases:

01

Estudio de las categorías OWASP más relevantes

02

Identificación de vulnerabilidades reales (CVE) y propuesta de contramedidas

Objetivo: proteger la confidencialidad, integridad y disponibilidad de los sistemas financieros.

Metodología

Análisis teórico

Revisión de las 10 categorías de vulnerabilidades OWASP.

Investigación aplicada

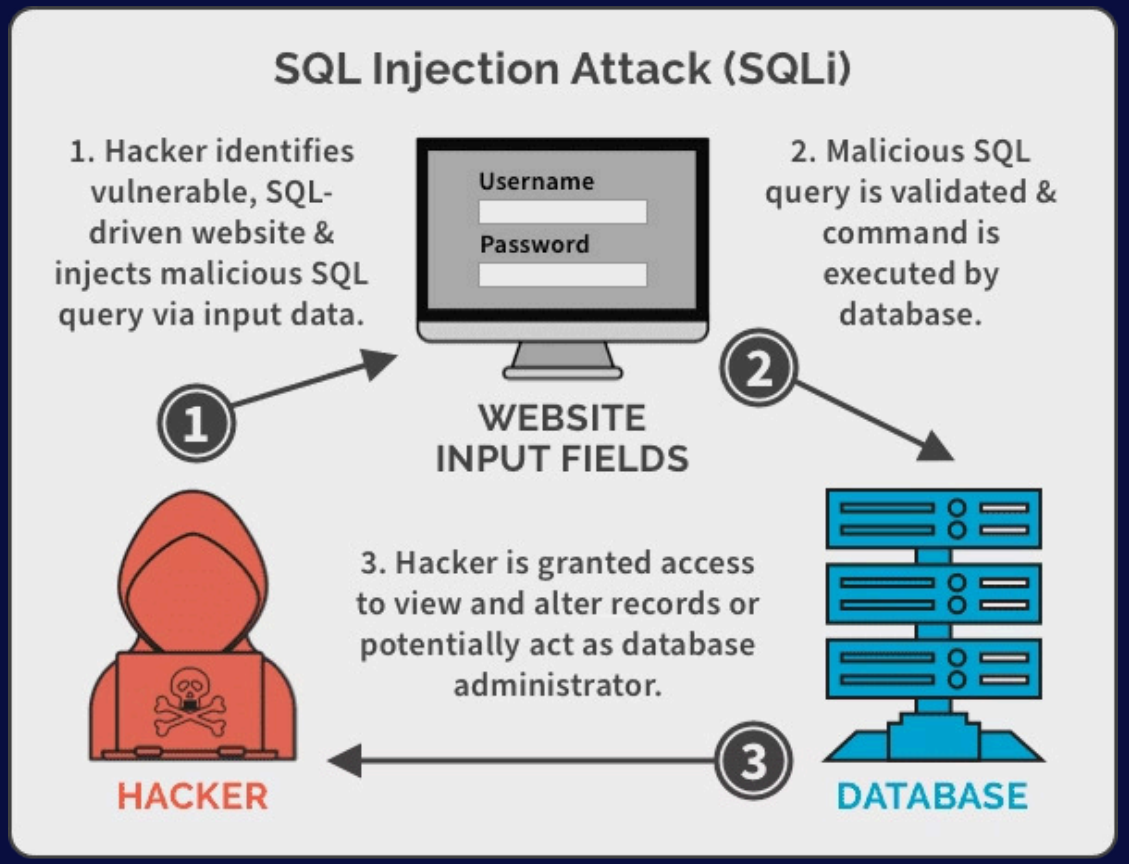
Recopilación de CVEs recientes (2025) en fuentes como NVD, Cloudflare, Unit42.

Evaluación de impacto

Cada vulnerabilidad fue clasificada por severidad y contexto financiero.

Diseño de contramedidas

Basadas en estándares como ISO 27001, NIST y GDPR.



Inyección (Injection)

- Permite que datos no validados se ejecuten directamente en un intérprete (SQL, shell, etc.).
- **Ejemplo:** *SQL Injection* o *Command Injection*.
- Impacto: acceso o modificación de información sensible.
- **CVE destacado:** CVE-2025-24813 (Apache Camel RCE).
- Relevancia: riesgo de ejecución remota en entornos financieros.

Autenticación rota (Broken Authentication)

El problema

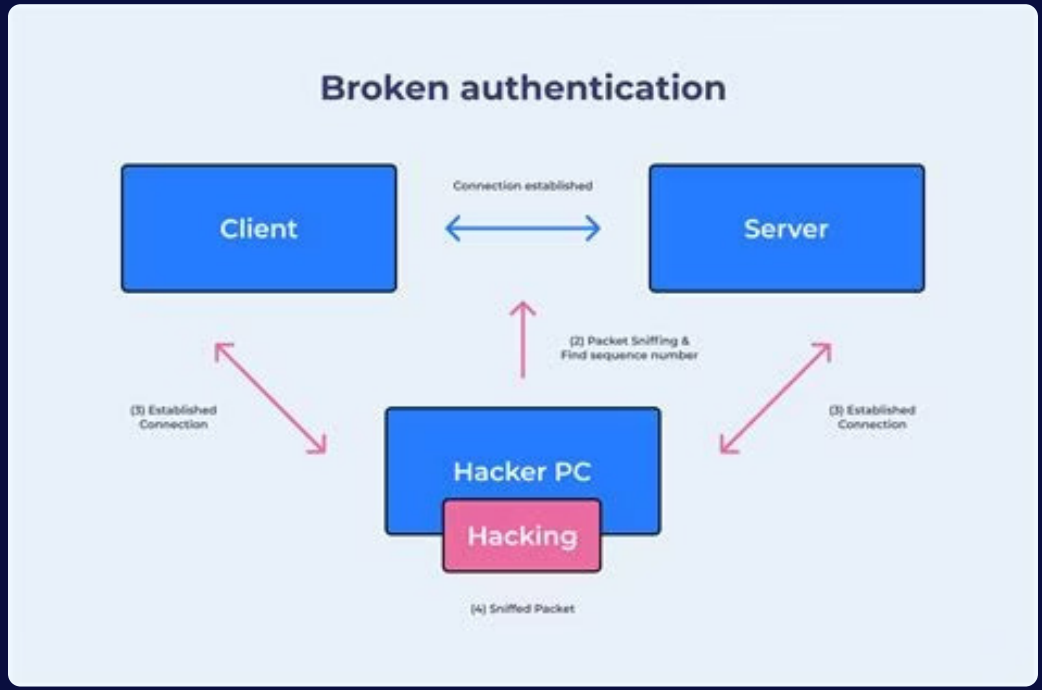
Fallos en los mecanismos de inicio de sesión o tokens permiten acceso no autorizado.

Consecuencias: suplantación de identidad, robo de cuentas.

CVE ejemplo: CVE-2025-49704 (bypass de autenticación en SharePoint).

La solución

Recomendación: uso de **MFA**, gestión segura de contraseñas y bloqueo progresivo.



Exposición de datos sensibles

Protección inadecuada

Protección inadecuada de datos personales o financieros.

Casos comunes

Falta de HTTPS, cifrado débil, exposición en logs.

Riesgo directo

Riesgo directo para la confidencialidad del cliente.

Contramedida: cifrado AES-256 y TLS 1.3, cumplimiento con **GDPR e ISO 27001**.

Attaxion

Potential Impacts of Sensitive Data Exposure

01	02	03	04	05
				
Regulatory violation	Legal costs	Reputational damage	Operational disruption	Customer churn

Control de acceso roto

Usuarios acceden a recursos restringidos manipulando parámetros o URLs.

CVE ejemplo

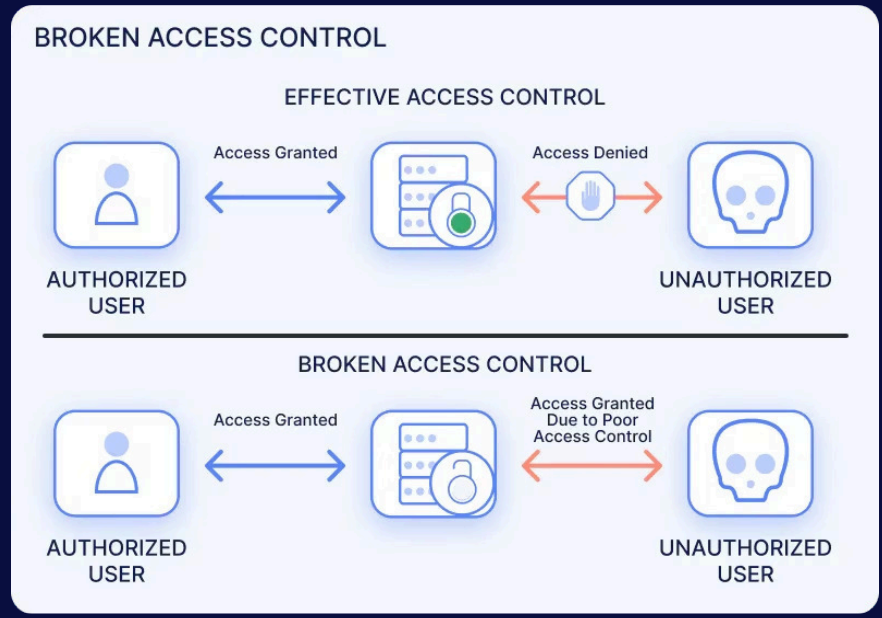
CVE-2025-29927 (bypass de autorización en Next.js).

Impacto

Exposición de información crítica.

Solución

Implementar controles RBAC/ABAC y verificación en backend.



Configuración de seguridad incorrecta

> Configuraciones por defecto o servicios innecesarios activos.
>
> Uno de los fallos **más comunes y explotables**.

Soluciones:

Revisar configuraciones periódicamente

Eliminar credenciales por defecto

Aplicar server hardening

Cross-Site Scripting (XSS)

Inyección de JavaScript malicioso en el navegador del usuario.

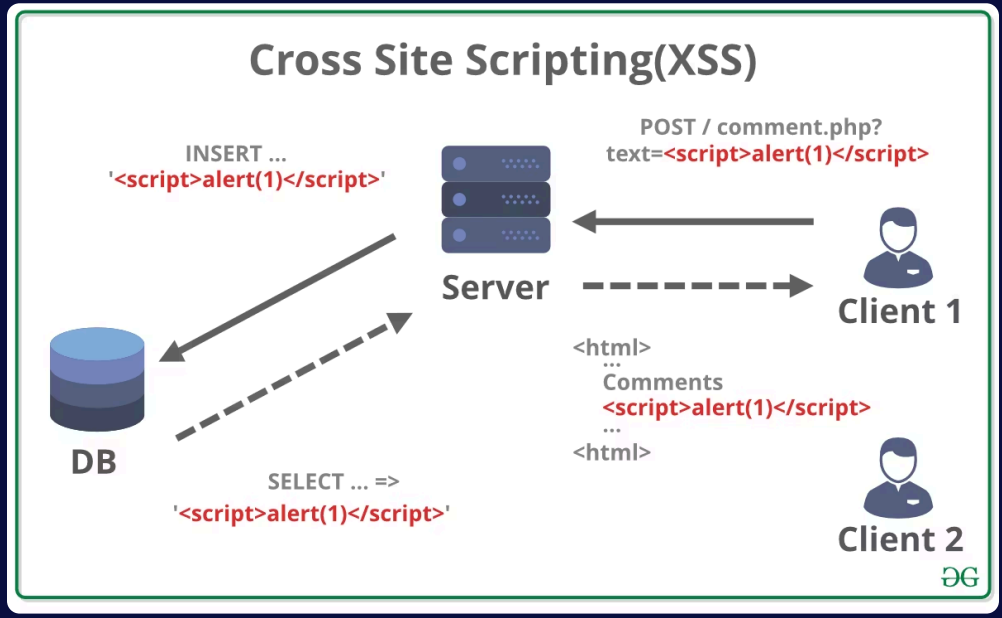
Ejemplos y consecuencias

Ejemplos: CVE-2025-27448 (XSS en dashboards) y **CVE-2025-4123** (Grafana).

Consecuencias: robo de cookies, redirección a sitios falsos.

Medidas de protección

- Escapar salidas HTML/JS.
- Usar frameworks seguros (React, Angular).
- Implementar **Content Security Policy (CSP)**.



Deserialización insegura

Datos manipulados pueden ejecutar código malicioso al deserializarse.

CVE destacados	Impacto
CVE-2025-53770 (SharePoint) y CVE-2025-23006 (SMA1000 Appliance).	Ejecución remota de comandos (RCE).

Contramedidas:

- No deserializar datos no confiables.
- Usar formatos validados (JSON).
- Firmas digitales en objetos.

Uso de componentes vulnerables

La reutilización de librerías y frameworks con vulnerabilidades conocidas puede ser un punto ciego crítico en la seguridad de las aplicaciones web.

Riesgo principal

Dependencia de software desactualizado y con fallos de seguridad documentados.

Ejemplo: CVE-2025-24813 (RCE crítica en Apache Camel).

Solución 1: SBOM

Mantener un **Software Bill of Materials (SBOM)** para inventariar todos los componentes.

Solución 2: SCA

Actualizar dependencias de forma proactiva con análisis de composición de software (**SCA**).

Registro y monitoreo insuficientes

La falta de trazabilidad y visibilidad en los sistemas de registro impide detectar ataques a tiempo.

Consecuencias:

- Respuesta tardía o ineficaz ante incidentes de seguridad.
- Redirección a URLs manipuladas por el atacante.
- Facilita **phishing** o fuga de sesión.
- **Ejemplo:** CVE-2025-4123 (Grafana).
- Prevención:
  - Validar URLs antes de redirigir.
  - Usar listas blancas.
  - Mostrar advertencias al salir del dominio.
- Dificultad para realizar análisis forenses.
- Incumplimiento de regulaciones y normativas (GDPR, PCI DSS).

Contramedidas recomendadas:

Implementar un SIEM

Para consolidar, analizar y correlacionar eventos de seguridad de diversas fuentes.

Alertas automáticas

Generar notificaciones en tiempo real ante anomalías o actividades sospechosas.

Logs seguros y auditables

Asegurar la integridad y disponibilidad de los registros para análisis forense.

Redirecciones y reenvíos inseguros

Un atacante puede manipular la URL a la que la aplicación redirige al usuario, llevando a sitios maliciosos o revelando información sensible.

Impacto: Facilita ataques de phishing, robo de credenciales o fuga de sesiones activas.
CVE destacado: CVE-2025-4123 (Vulnerabilidad de redirección en Grafana que permitía XSS).

Prevención:

01

Validación estricta de URLs

Asegurarse de que todas las URLs de redirección sean validadas rigurosamente en el servidor y que apunten solo a destinos permitidos.

02

Uso de listas blancas

Implementar una lista blanca de dominios y rutas a las que la aplicación puede redirigir, rechazando cualquier otra petición.

03

Advertencias al usuario

Proporcionar una advertencia explícita al usuario antes de redirigirlo a un dominio externo o no verificado.

VULNERABILIDAD OPEN REDIRECT

ESGEEKS

Resumen de CVEs Críticos

A continuación se presenta una tabla resumen de las vulnerabilidades más críticas identificadas, destacando su severidad y el principal riesgo asociado para el entorno financiero.

CVE-2025-53770	Deserialización insegura	9.8	Ejecución remota (RCE)
CVE-2025-29927	Control de acceso roto	9.1	Acceso indebido
CVE-2025-24813	Inyección / RCE	10.0	Ejecución remota
CVE-2025-49704	Autenticación rota	8.8	Bypass de seguridad
CVE-2025-27448	XSS	6.8	Inyección JavaScript

Estos CVEs representan los riesgos más significativos analizados, con potencial impacto directo en la confidencialidad, integridad y disponibilidad de los activos críticos.