Advertencia

Técnica

Buenas prácticas para protección de portales web

COLCERT AD-20260626- 035

TLP:CLEAR

Resumen Ejecutivo

Los portales web del Estado son un blanco frecuente de defacement, filtración de datos e indisponibilidad. Estas amenazas evolucionan constantemente y afectan de manera permanente los servicios digitales que soportan la atención a la ciudadanía y la operación de las entidades públicas. La mayoría de los incidentes conocidos explotan un conjunto reducido de vulnerabilidades ampliamente documentadas y mitigables mediante buenas prácticas de seguridad. **Este boletín presenta, en un lenguaje claro y práctico, los diez riesgos más críticos** para las aplicaciones web definidos en la referencia internacional más reconocida para la identificación y priorización de riesgos de seguridad en aplicaciones web - OWASP Top 10:2025. Asimismo, ofrece recomendaciones técnicas dirigidas a administradores de sistemas y equipos técnicos y de desarrollo y responsables de seguridad para fortalecer la protección de los portales institucionales. Cada riesgo se aborda a través de tres aspectos clave**: qué es, por qué representa un riesgo y qué acciones deben implementarse para mitigarlo.**

NIVEL DE RIESGO

ALTO

Panorama: los 10 riesgos de un vistazo

Código	Riesgo	Prioridad
A01	Control de acceso roto	CRÍTICA
A02	Configuración de seguridad incorrecta	CRÍTICA
A03	Fallas en la cadena de suministro de software	ALTA
A04	Fallas criptográficas	ALTA
A05	Inyección	CRÍTICA
A06	Diseño inseguro	ALTA
A07	Fallas de autenticación	ALTA
A08	Fallas de integridad de software o datos	MEDIA
A09	Fallas de registro y alertamiento	MEDIA
A10	Manejo inadecuado de condiciones excepcionales	MEDIA

Impacto para Colombia y la región

- **Concentración de servicios y datos.** Los portales del Estado concentran trámites, datos personales y servicios críticos; un incidente afecta la confianza ciudadana y la continuidad del servicio.
- **Coyunturas de alta visibilidad.** En escenarios de alta exposición pública, la disponibilidad, integridad y confiabilidad de los portales institucionales adquieren una importancia estratégica. Incidentes como el defacement, la indisponibilidad de servicios y las campañas de desinformación pueden afectar la confianza de la ciudadanía y la continuidad de los servicios digitales del Estado.
- **Riesgos dominantes.** Las tres familias más críticas: control de acceso roto, configuración incorrecta e inyección y toma de control observadas en entidades públicas.
- **Marco normativo.** La exposición de datos personales activa obligaciones de la Ley 1581 de 2012 y la notificación a la SIC; los incidentes de alto perfil aumentan el riesgo de phishing institucional e ingeniería social.
- **Prioridad transversal.** Priorizar HTTPS/HSTS, MFA para administradores, gestión de la superficie de ataque (ASM) y monitoreo con alertamiento en tiempo real.

Recomendaciones técnicas y de mitigación

A01 Control de acceso roto — CRÍTICA

¿Qué es? *Un usuario hace o ve algo que no le corresponde:* accede a registros de otra persona, llega a funciones de administrador o se salta restricciones. Sigue siendo el riesgo #1, presente en cerca del 3,7 % de las aplicaciones probadas; en esta versión absorbe también el SSRF (peticiones del servidor forzadas hacia destinos no permitidos).

¿Por qué importa? Es la puerta de entrada más común a filtraciones masivas y a la toma de control de un portal.

Un caso típico: cambiar el número de identificación en la URL y obtener los datos de otro ciudadano (IDOR).

¿Qué hacer?

- ❑ Denegar por defecto y aplicar mínimo privilegio: nadie accede a nada salvo que se autorice explícitamente.
- ❑ Verificar la autorización en el servidor en cada solicitud y a nivel de objeto; nunca confiar en controles del lado del navegador.
- ❑ Centralizar la lógica de permisos en un único componente reutilizable.
- ❑ Invalidar sesiones en el servidor al cerrar sesión; tokens con expiración corta.
- ❑ Contra SSRF: validar y usar lista blanca de destinos en cualquier función que haga peticiones salientes.
- ❑ Registrar y alertar los accesos denegados repetidos.

A02 Configuración de seguridad incorrecta — CRÍTICA

¿Qué es? *Servicios, servidores o frameworks desplegados con ajustes inseguros:* credenciales por defecto, paneles expuestos, directorios listables, cabeceras que revelan versiones o errores que muestran detalles internos. Subió del puesto 5 al 2.

¿Por qué importa? *Una configuración descuidada entrega información y accesos al atacante sin necesidad de explotar código.* Un panel de administración accesible desde Internet o un bucket de almacenamiento con permisos mal configurados bastan para comprometer el portal.

¿Qué hacer?

- ❑ Aplicar hardening: eliminar cuentas y contraseñas por defecto, páginas de ejemplo y listados de directorio.
- ❑ Añadir cabeceras de seguridad (HSTS, CSP, X-Content-Type-Options, X-Frame-Options) y ocultar las que revelan versiones.
- ❑ Mensajes de error genéricos al usuario; el detalle va solo al log.
- ❑ No exponer a Internet paneles de administración, bases de datos ni servicios de gestión.
- ❑ Gestionar la configuración de forma repetible (IaC) y revisarla periódicamente.

A03 Fallas en la cadena de suministro de software — ALTA

¿Qué es? *Categoría ampliada (antes 'componentes vulnerables y desactualizados').* Cubre todo el ecosistema: dependencias de terceros, librerías, plugins de CMS, herramientas de compilación y pipelines de CI/CD. Tiene los puntajes de explotación e impacto más altos del listado.

¿Por qué importa? *Un solo componente comprometido* - una librería, un plugin de WordPress, un paquete malicioso - puede infectar todo el portal.

¿Qué hacer?

- ❑ Mantener un inventario de dependencias (SBOM) y usar análisis de composición (SCA) para detectar componentes vulnerables.
- ❑ Fijar versiones y verificar la integridad (hashes/firmas) de los paquetes que se instalan.
- ❑ Asegurar el pipeline CI/CD: proteger secretos, accesos y runners; usar solo repositorios confiables.
- ❑ Evaluar proveedores y librerías de terceros antes de incorporarlos.

Recomendaciones Técnicas y de Mitigación

A04 Fallas criptográficas — ALTA

¿Qué es? Protección débil o ausente de los datos: tráfico sin cifrar, contraseñas mal protegidas, algoritmos obsoletos o llaves mal gestionadas. Afecta a cerca del 3,8 % de las aplicaciones.

¿Por qué importa? Lleva directo a la exposición de datos sensibles (cédulas, credenciales, información personal). Si las contraseñas se guardan en texto claro o con MD5, una sola filtración las compromete todas.

¿Qué hacer?

- ❑ Forzar HTTPS en todo el portal y activar HSTS; deshabilitar cifrados débiles.
- ❑ Cifrar los datos sensibles en reposo y clasificar qué requiere protección.
- ❑ Hash de contraseñas con algoritmos fuertes (bcrypt, argon2); nunca MD5/SHA1 ni texto claro.
- ❑ Gestionar llaves con rotación y almacenamiento seguro (KMS/HSM); jamás en el código.
- ❑ No exponer datos sensibles en URLs, logs ni cachés.

A05 Inyección — CRÍTICA

¿Qué es? *El atacante introduce datos maliciosos que la aplicación interpreta como comandos:* SQL Injection, Cross-Site Scripting (XSS), inyección de comandos. Una de las categorías con más vulnerabilidades registradas.

¿Por qué importa? Permite robar o alterar la base de datos, secuestrar sesiones o ejecutar código. El XSS es muy frecuente y se usa para defacement y robo de credenciales.

¿Qué hacer?

- ❑ Consultas parametrizadas u ORM; nunca concatenar entrada del usuario en SQL.
- ❑ Validar y sanear toda la entrada en el servidor con listas blancas.
- ❑ Codificar la salida según el contexto (HTML, JS, URL) para prevenir XSS; CSP como apoyo.
- ❑ Mantener un Web Application Firewall (WAF) como una capa complementaria de protección.
- ❑ Incluir pruebas SAST y DAST en el ciclo.

A06 Diseño inseguro — ALTA

¿Qué es? *Fallas que nacen en el diseño, no en la implementación:* faltan controles pensados desde el inicio (límites, validaciones, flujos seguros). No se corrige solo con parches.

¿Por qué importa? Si una funcionalidad se diseñó sin pensar en el abuso (recuperación de contraseña débil, un proceso sin límites) el problema persiste aunque el código esté 'bien escrito'.

¿Qué hacer?

- ❑ Modelado de amenazas desde el diseño e 'historias de abuso' junto a los requisitos.
- ❑ Reutilizar patrones seguros y definir límites de tasa y de recursos por diseño.
- ❑ Separar entornos y datos según su sensibilidad.
- ❑ Exigir requisitos de seguridad explícitos en cada funcionalidad nueva.

A07 Fallas de autenticación — ALTA

¿Qué es? *Debilidades al verificar la identidad*: contraseñas débiles, ausencia de segundo factor, sesiones mal gestionadas o poca protección contra fuerza bruta. Antes 'Identificación y autenticación'.

¿Por qué importa? Habilita el robo de cuentas, sobre todo de administradores. El credential stuffing (probar credenciales filtradas) compromete portales que no limitan los intentos.

¿Qué hacer?

- ❑ Implementar MFA, en especial para administradores y back-office.
- ❑ Bloquear o ralentizar los intentos ante fuerza bruta y credential stuffing.
- ❑ Política de contraseñas robusta y verificación contra credenciales filtradas.
- ❑ Cookies HttpOnly/Secure/SameSite, con expiración y rotación del identificador de sesión.
- ❑ No revelar si un usuario existe; evitar preguntas de seguridad débiles.

A08 Fallas de integridad de software o datos — MEDIA

¿Qué es? Falta de verificación de que el software, el código o los datos no han sido alterados: actualizaciones sin firmar, deserialización de datos no confiables o artefactos modificados.

¿Por qué importa? *Permite introducir código malicioso vía una actualización o un objeto manipulado.* Un plugin actualizado desde una fuente no verificada puede abrir la puerta.

¿Qué hacer?

- ❑ Verificar firmas e integridad antes de desplegar.
- ❑ Proteger el pipeline e incorporar revisión de código.
- ❑ No deserializar datos de fuentes no confiables sin validar.
- ❑ Actualizar plugins/CMS solo desde fuentes verificadas.

A09 Fallas de registro y alertamiento — MEDIA

¿Qué es? *No registrar los eventos relevantes o registrarlos sin alertar.* El nombre enfatiza el a**lertamiento**: un buen log sin alertas sirve de poco.

¿Por qué importa? Sin detección a tiempo, un atacante puede operar meses sin ser visto. Aquí se pierde la oportunidad de detectar un incidente.

¿Qué hacer?

- ❑ Registrar logins, fallos, accesos denegados y cambios administrativos.
- ❑ Centralizar en un SIEM y ALERTAR en tiempo real.
- ❑ Sincronizar relojes en UTC y proteger los logs frente a manipulación.
- ❑ Definir runbooks de respuesta y probar la detección.
- ❑ Establecer una retención acorde con la normativa.

A10 Manejo inadecuado de condiciones excepcionales — MEDIA

¿Qué es? *Categoría nueva en 2025. Reúne el manejo incorrecto de errores y situaciones anómalas:* fallos que abren en vez de cerrar, errores lógicos y comportamientos ante condiciones límite.

¿Por qué importa? Cuando un sistema falla, debe hacerlo de forma segura, si un error otorga acceso por defecto ('fail open') o expone detalles internos, el fallo se convierte en una vulnerabilidad.

¿Qué hacer?

- ❑ Manejar los errores con 'fail closed': un fallo nunca debe otorgar acceso ni privilegios.
- ❑ No exponer detalles internos en los errores; mensaje genérico al usuario, detalle al log.
- ❑ Controlar tiempos de espera, reintentos y límites para evitar denegación de servicio.
- ❑ Probar explícitamente las rutas de error y los casos límite, no solo el 'camino feliz'.

COLCERT

Lista rápida de verificación

Recomendaciones mínimas que se deberían aplicar.

- ✓ HTTPS + HSTS en todo el portal y cabeceras de seguridad activas.
- ✓ MFA obligatorio para administradores y back-office.
- ✓ Consultas parametrizadas y validación de entrada en el servidor.
- ✓ Denegar por defecto y verificar autorización por objeto en cada solicitud.
- ✓ Retirar de Internet los paneles de administración y servicios de gestión.
- ✓ SCA y actualización disciplinada de CMS, plugins y dependencias.
- ✓ Logs centralizados con alertas en tiempo real, no solo almacenamiento.
- ✓ Copias de respaldo probadas y plan de respuesta a incidentes vigente.

Referencia y contacto

Ante un incidente de seguridad, las entidades del Estado pueden contactar al **ColCERT** y activar su plan de respuesta a través de los canales oficiales: www.colcert.gov.co

¿Cómo fortalecer la seguridad de los portales?

El ColCERT, a través de su línea de gestión de vulnerabilidades y gestión de superficie de ataque (ASM), puede evaluar el grado de exposición de los porxtales de su entidad frente al OWASP Top 10:2025, identificar las desviaciones priorizadas por criticidad y entregar un informe técnico con recomendaciones de remediación. Las entidades del Estado pueden solicitar este acompañamiento a través de los canales oficiales del ColCERT.

Fuentes

OWASP Foundation. OWASP Top 10:2025. Disponible en: https://owasp.org/Top10/2025