Top 10 OWASP

Alejandro Álvarez Cerviño
Alejandro Cabrera Mateo
Robert Carmona García
Aketza Cítores Franco
Enrique Curt Moscoso
Adrián Del Valle Arroyo
Manuel Díaz de la Rocha de Castro
José María Fernández Salcedo
Pablo Granado Borga
Ainhoa Moreno Ruiz
Jose Luis Venega Sánchez
Luis Taracena Naranjo
Adrián Vaca Suano
Angel Custodio Ruiz Tejero
Pablo Troncoso Zambrano

Broken Access Control:

La vulnerabilidad Broken Access Control ocurre cuando una falla o una ausencia de mecanismos de control de acceso, por ejemplo, Autenticación (verificar la identidad del usuario) o Autorización (comprobar si el usuario tiene permiso de acceder a un recurso) le permite a un usuario acceder a un recurso que está fuera de sus permisos previstos.

Cryptographic Failures:

Son vulnerabilidades o errores en los mecanismos de cifrado que comprometen la seguridad de los datos. Estas fallas pueden ocurrir por diversas razones, desde una implementación incorrecta hasta el uso de algoritmos débiles o mal configurados.

Las consecuencias incluyen el robo de información o identidad. Para prevenirlas, es esencial usar algoritmos actualizados, gestionar correctamente las claves, configurar adecuadamente los sistemas y aplicar buenas prácticas de seguridad.

Inyección:

Ocurren cuando los atacantes explotan las vulnerabilidades de una aplicación para enviar código malicioso a un sistema. Este tipo de exploit puede permitirles ejecutar comandos no autorizados, acceder a datos o manipular las operaciones del sistema.

Debido a que los ataques de inyección pueden ser muy peligrosos, y debido a que su uso está muy extendido, representan una amenaza crítica para la ciberseguridad en la actualidad.

Insecure Design

El diseño inseguro es una categoría amplia que representa diferentes debilidades, expresadas como “diseño de control ineficaz o falta de control”. El diseño inseguro no es la fuente de todas las demás categorías de riesgo principales. Existe una diferencia entre el diseño inseguro y la implementación insegura. Diferenciamos entre fallas de diseño y defectos de implementación por una razón: tienen diferentes causas y soluciones. Un diseño seguro puede tener defectos de implementación que generan vulnerabilidades que pueden ser explotadas. Un diseño inseguro no puede solucionarse con una implementación perfecta ya que, por definición, nunca se crearon los controles de seguridad necesarios para defenderse de ataques específicos. Uno de los factores que contribuyen al diseño inseguro es la falta de un perfil de riesgo empresarial inherente al software o sistema que se está desarrollando y, por lo tanto, la imposibilidad de determinar qué nivel de diseño de seguridad se requiere.

Configuración incorrecta de seguridad

El 90% de las solicitudes probadas muestran configuraciones incorrectas, con una incidencia del 4% y más de 208,000 ocurrencias de debilidades comunes (CWE), como CWE-16 (Configuración) y CWE-611 (Restricción inadecuada de referencias de entidad externa XML). Las vulnerabilidades incluyen configuraciones incorrectas de seguridad, funciones innecesarias habilitadas, cuentas predeterminadas activas, manejo inadecuado de errores y software desactualizado. Para prevenir, se recomienda un proceso de instalación seguro, eliminación de funciones innecesarias, revisión de configuraciones, segmentación de aplicaciones y automatización para verificar configuraciones seguras.

Componentes Vulnerables y obsoletos

Este riesgo se refiere a la práctica de utilizar componentes de terceros(como bibliotecas, marcos de trabajo, etc.) que tienen vulnerabilidades conocidas o que ya no están siendo mantenidos activamente. Eres vulnerable a este tipo de ataque si usas software desactualizado, los desarrolladores no comprueban o parchean sus programas o no escaneas en busca de vulnerabilidades. Pero puedes protegerte actualizando con frecuencia los programas, eliminando dependencias o librerías innecesarias y obtienes los programas solo por fuentes oficiales.

Fallas de identificación y autentificación

Previamente denominada como *Pérdida de Autenticación*, descendió desde la segunda posición, y ahora incluye CWEs que están más relacionados con fallas de identificación.La confirmación de identidad y autenticación toman gran importancia por lo que un fallo en esta crea una gran debilidad, permitiendo ataques como:ataques automatizados como la reutilización de credenciales,ataques de fuerza bruta etc.

Software and Data Integrity Failures

Los fallos de integridad en software y datos ocurren cuando el código o la infraestructura no están protegidos adecuadamente contra alteraciones. Algunos ejemplos, incluyen el uso de plugins o bibliotecas de fuentes no confiables, pipelines CI/CD inseguros que permiten accesos no autorizados o la inyección de código malicioso, y actualizaciones automáticas sin verificaciones de integridad

Fallas en el Registro y Monitoreo

Las **Fallas en el Registro y Monitoreo** (A09:2021, OWASP) ocurren cuando un sistema no registra ni monitorea adecuadamente eventos de seguridad, lo que impide detectar ataques o comportamientos anómalos. Esto puede resultar en incidentes no detectados, dificultar la investigación posterior a un ataque y comprometer la capacidad de respuesta rápida. La falta de monitoreo puede exponer a la organización a riesgos de seguridad prolongados y cumplir mal con normativas legales.

SSRF

Esta vulnerabilidad ocurre cuando una aplicación web permite hacer consultas HTTP del lado del servidor hacia un dominio arbitrario elegido por el atacante.Esto le permite a un atacante hacer conexión con servicios de la infraestructura interna donde se aloja la web y exfiltrar información sensible, por ejemplo los puertos abiertos internamente.