Protocolo de Seguridad en el Desarrollo

Este procedimiento establece las recomendaciones y prácticas más adecuadas que deben seguirse en todas las etapas del desarrollo de software: desde la definición de requisitos hasta el mantenimiento en producción. Su propósito es reducir al mínimo las vulnerabilidades identificadas para proteger la confidencialidad y la integridad de los datos almacenados y garantizar que el proceso de desarrollo se lleve a cabo de manera sólida y conforme a estándares reconocidos internacionalmente como OWASP o ISO/IEC 27001.

Incorporando la seguridad en cada etapa del proceso de desarrollo (Seguridad desde el Diseño).
Reducir los riesgos a través de la detención y mitigación anticipada de vulnerabilidades.
Garantizar que la aplicación y los datos de los usuarios se mantengan confidenciales y seguros y estén disponibles cuando se necesiten.
Seguir las regulaciones y normas de seguridad ampliamente aceptadas.

Requisitos y Evaluación de Riesgos.

Establecimiento de los Requisitos de Seguridad: Incorporación de condiciones de seguridad detalladas desde la etapa inicial de recolección de requisitos.
Análisis de Riesgos de Seguridad: Llevar a cabo una evaluación de riesgos (por ejemplo utilizando OWASP Threat Modeling), para identificar posibles puntos vulnerables e implementar medidas preventivas.

Principios de diseño seguro - Implementar “Seguridad por Diseño” y “Defensa en Profundidad".
Segregación de Ambientes : La práctica de mantener ambientes distintos para desarrollo, pruebas y producción resulta en una menor exposición de datos sensibles.
Revisión de Diseño : Es importante confirmar que el diseño arquitectónico incluya métodos de autenticación y autorización adecuados, cifrado seguro y gestión efectiva de errores .

Prácticas de codificación segura recomendadas, siguiendo las directrices de OWASP y aplicando patrones de diseño para evitar vulnerabilidades comunes.
Implementar medidas rigurosas para validar y limpiar todas las entradas del usuario, asegurando la validación y sanitización de dichos datos.
Mantenimiento de Dependencias es clave para asegurar que las librerías y frameworks estén al día y emplear herramientas para examinar vulnerabilidades en las dependencias utilizadas.

Revisar código y realizar análisis estáticamente regularmente, además de emplear herramientas como SonarQube, son prácticas recomendables para detectar potenciales problemas de seguridad.

Pruebas de seguridad en una aplicación requieren la realización de pruebas de penetración y escaneos automatizados para identificar posibles vulnerabilidades.
Integrar pruebas de seguridad en el proceso de integración y despliegue continuo es crucial para detectar problemas de manera anticipada en el ciclo de desarrollo de software.
Análisis Dinámico - Mejorar el análisis estática mediante pruebas de seguridad durante la ejecución del programa.

Configuración de seguridad ópticima es implementar ajustes seguros en servidores, contenedores y servicios, conforme al principio de ``Privilegio Mínimo'' y a los modelos de “Confianza Cero”.
Seguridad en la Contenerización : Emplear imágenes Docker simples, realizar escaneos de manera periódica y manejar las vulnerabilidades mediante registros de contenedores.
Controles sólidos de acceso y supervisión son fundamentales para garantizar la seguridad de los sistemas informáticos mediante la implementación de autenticación multifactorial.

Mantenimiento y revisión
Mantenimiento y Mejoras: Es importante llevar a cabo actualizaciones de manera regular y aplicar parches de seguridad en cuanto se detecte alguna vulnerabilidad.
Auditorías y Revisiones en Seguridad: Organizar revisiones internas y externas para evaluar la eficiencia del sistema implementado.
Plan de Respuesta a Incidentes: Crear pautas precisas para identificar y manejar de manera efectiva eventos de seguridad.