

**GASIC 8– Seguridad en el Desarrollo y Adquisición de Software**

**Preguntas de Auditoría**

**Herramientas y Cadenas de Soporte**

**Competencias Básicas del Auditor**

1. **Conocimiento Profundo en Desarrollo de Software Seguro:** Entender los principios de seguridad en el desarrollo de software, incluyendo buenas prácticas de codificación, diseño seguro, y consideraciones de seguridad en el ciclo de vida del desarrollo.
2. **Experiencia en Identificación de Vulnerabilidades:** Tener habilidades para identificar y evaluar vulnerabilidades en el código y el diseño del software, incluyendo análisis estático y dinámico, pruebas de penetración y revisión de arquitecturas.
3. **Conocimiento en Criptografía y Protección de Datos:** Comprender los principios de la criptografía y cómo se aplica en la protección de datos sensibles dentro del software, incluyendo la gestión segura de claves y la integridad de los datos.
4. **Habilidades en Pruebas de Seguridad y Evaluación de Riesgos:** Ser capaz de realizar pruebas de seguridad exhaustivas para evaluar la resistencia del software frente a posibles ataques, así como evaluar los riesgos asociados con el software.
5. **Conocimientos en Cumplimiento y Normativas:** Estar familiarizado con los requisitos de cumplimiento y normativas relevantes en el ámbito de la seguridad del software, como GDPR, PCI DSS, y normas ISO/IEC.
6. **Capacidad Analítica:** Ser capaz de analizar datos de seguridad y tendencias para identificar posibles amenazas y vulnerabilidades emergentes en el software.
7. **Habilidades de Comunicación Técnica:** Ser capaz de comunicar de manera efectiva los hallazgos de seguridad a los desarrolladores, gerentes y otros interesados, y proponer recomendaciones claras y prácticas.
8. **Conocimientos en Gestión de Incidentes y Respuesta a Incidentes:** Entender los procedimientos y prácticas para la gestión de incidentes de seguridad en el software, incluyendo la capacidad de respuesta rápida y eficaz.

# Visión General

La selección, integración y gestión de herramientas dentro de la cadena de soporte del desarrollo de no solo automatizan procesos clave, sino que también aseguran la trazabilidad, auditabilidad y resiliencia frente a vulnerabilidades. Desde una perspectiva de auditoría, es crucial que las organizaciones cuenten con procesos claros para evaluar, configurar y monitorear las herramientas utilizadas en el ciclo de vida del desarrollo de software (SDLC).

**1.1 Definición e Integración de Herramientas**

Las cadenas de herramientas deben definirse categorizando los tipos de herramientas obligatorias para cada etapa del desarrollo, incluyendo aquellas destinadas a la seguridad, como escáneres de vulnerabilidades o herramientas de análisis de código. La integración de estas herramientas debe contemplar qué información se compartirá entre ellas y en qué formatos, asegurando la interoperabilidad y la eficiencia operativa. Un auditor evaluará si las herramientas seleccionadas están alineadas con los objetivos de seguridad y si su integración ha sido documentada adecuadamente.

**1.2 Evaluación y Configuración de Herramientas**

Antes de su implementación, las herramientas deben ser evaluadas tanto en términos de funcionalidad como de seguridad, verificando su capacidad para cumplir con los estándares establecidos. Esto incluye la configuración basada en código para garantizar la consistencia en su implementación y el uso de tecnologías que permitan compilaciones reproducibles. Desde el punto de vista de la auditoría, es importante revisar la documentación de las evaluaciones realizadas y verificar que las herramientas cumplen con los requisitos de seguridad antes de ser integradas.

**1.3 Monitoreo y Actualización de Herramientas**

La supervisión continua de las herramientas es esencial para detectar problemas operativos o vulnerabilidades de seguridad. Además, es necesario verificar periódicamente la integridad y procedencia de cada herramienta, asegurándose de que se mantengan actualizadas y que cualquier vulnerabilidad identificada sea abordada oportunamente. Un auditor revisará los registros de supervisión y las políticas de actualización implementadas para garantizar que las herramientas permanecen seguras y funcionales a lo largo del tiempo.

**1.4 Auditabilidad y Gestión de Artefactos**

Las herramientas deben generar rastros de auditoría que documenten acciones relacionadas con la seguridad del desarrollo, asegurando la trazabilidad de cada actividad. Es fundamental establecer políticas claras sobre la retención de datos de artefactos y asignar responsabilidades específicas para su creación cuando las herramientas no puedan generarlos automáticamente. Desde la perspectiva de la auditoría, estos rastros son esenciales para evaluar el cumplimiento de las políticas de seguridad y la eficacia de las herramientas en la cadena de soporte.

**1.5 Automatización y Orquestación de la Cadena de Herramientas**

La automatización de la gestión y orquestación de herramientas permite optimizar los procesos del SDLC, minimizando errores y mejorando la consistencia en la implementación de prácticas de seguridad. Las tecnologías automatizadas deben ser evaluadas por su capacidad para garantizar la resiliencia de la cadena de herramientas y facilitar la auditoría. Un auditor verificará que la automatización se haya implementado de acuerdo con las políticas organizacionales y que contribuya a la mejora continua de la seguridad en el desarrollo de software.

# Modelo de Madurez

|  |  |
| --- | --- |
| NIVEL 1 [Impredecible y reactivo]: La actividad se completa, pero a menudo se retrasa y supera el presupuesto. | 1. Definir categorías de cadenas de herramientas y especificar las herramientas obligatorias o tipos a utilizar. 2. Identificar herramientas de seguridad para integrarlas en la cadena de herramientas del desarrollador. 3. Definir qué información se pasará entre herramientas y qué formatos de datos se utilizarán. 4. Evaluar las capacidades de firma de las herramientas para crear registros/registros inmutables para la auditabilidad. 5. Utilizar tecnología automatizada para la gestión y orquestación de la cadena de herramientas. |
| NIVEL 2 [Gestionado a nivel de proceso]: Las actividades se planifican, ejecutan, miden y controlan. | 1. Evaluar, seleccionar y adquirir herramientas, y evaluar la seguridad de cada herramienta. 2. Integrar herramientas con otras herramientas y procesos existentes. 3. Utilizar una configuración basada en código para cadenas de herramientas. 4. Implementar tecnologías y procesos para compilaciones reproducibles. 5. Actualizar o reemplazar herramientas según sea necesario para abordar vulnerabilidades o agregar nuevas capacidades. 6. Supervisar continuamente las herramientas y registros para detectar problemas operativos y de seguridad. 7. Verificar periódicamente la integridad y procedencia de cada herramienta. |
| NIVEL 3 [Proactivo antes que reactivo]: Los estándares de toda la organización brindan orientación a través de proyectos, programas y carteras. | 1.Utilizar herramientas existentes para crear un seguimiento de auditoría de las acciones seguras relacionadas con el desarrollo. 2. Determinar la frecuencia de auditorías e implementar procesos necesarios. 3. Establecer políticas de seguridad y retención para datos de artefactos. 4. Asignar responsabilidad para crear artefactos cuando las herramientas no puedan generarlos. |
| NIVEL 4 [Gestionado Cuantitativamente]: La organización está basada en datos con objetivos cuantitativos de mejora del desempeño que son predecibles |  |
| NIVEL 5 [Optimizado]: La organización se centra en la mejora continua y está diseñada para girar y responder a las oportunidades y los cambios. |  |

1. **Preguntas de Auditoría para Herramientas y Cadenas de Soporte**

|  |
| --- |
| Temática: Definición y Gestión de Cadenas de Herramientas |
| La correcta definición y gestión de cadenas de herramientas es esencial para mantener un entorno de desarrollo de software seguro y eficiente. Esto implica seleccionar, integrar y gestionar herramientas que no solo faciliten el proceso de desarrollo, sino que también aseguren que se mantengan altos estándares de seguridad en cada etapa. Al evaluar estos controles, se puede identificar si las herramientas son adecuadas para proteger los activos digitales, garantizar la interoperabilidad y mantener la auditabilidad de las acciones realizadas.  Este levantamiento puede incluir análisis específicos tales como:   * ¿Se han definido claramente las categorías de cadenas de herramientas necesarias para el desarrollo y adquisición de software? * ¿Existen herramientas obligatorias especificadas que deben ser utilizadas en el proceso? * ¿Se evalúan y seleccionan herramientas basadas en criterios específicos de seguridad y funcionalidad? * ¿Se han identificado y seleccionado herramientas de seguridad adecuadas para integrarlas en la cadena de herramientas del desarrollador? * ¿Se integran estas herramientas de seguridad de manera efectiva con otras herramientas y procesos existentes? * ¿Qué información se pasa entre herramientas en la cadena de herramientas? * ¿Se han definido y estandarizado los formatos de datos utilizados para la comunicación entre herramientas? * ¿Las herramientas seleccionadas son compatibles con estos formatos de datos? * ¿Las herramientas utilizadas tienen capacidades de firma digital para crear registros inmutables? * ¿Se generan y mantienen registros detallados de todas las acciones realizadas por las herramientas? * ¿Estos registros son revisados periódicamente para asegurar su integridad y auditabilidad? * ¿Se utiliza tecnología automatizada para la gestión y orquestación de la cadena de herramientas? * ¿Existe un proceso formal para la evaluación, selección y adquisición de nuevas herramientas? * ¿Se evalúa regularmente la seguridad de cada herramienta utilizada? |

|  |
| --- |
| Temática: Definición y Gestión de Cadenas de Herramientas |
| La implementación y mantenimiento efectivos de herramientas en el desarrollo y adquisición de software son críticos para asegurar que los procesos sean seguros y eficientes. Es fundamental evaluar cómo se configuran, integran y actualizan estas herramientas para garantizar que se mantengan alineadas con los estándares de seguridad y que respondan adecuadamente a las amenazas emergentes. Esta evaluación ayuda a identificar áreas de mejora y asegura que las herramientas utilizadas proporcionen una protección continua y robusta.  Este levantamiento puede incluir análisis específicos tales como:   * ¿Se utiliza una configuración basada en código para las cadenas de herramientas? * ¿Las herramientas son actualizadas o reemplazadas según sea necesario para abordar vulnerabilidades o agregar nuevas capacidades? * ¿Existe un proceso establecido para la integración de herramientas con otras herramientas y procesos existentes? * ¿Se han implementado tecnologías y procesos para asegurar compilaciones reproducibles? * ¿Se supervisan continuamente las herramientas y registros para detectar problemas operativos y de seguridad? * ¿Se verifica periódicamente la integridad y procedencia de cada herramienta utilizada? * ¿Se utilizan herramientas existentes para crear un seguimiento de auditoría de las acciones seguras relacionadas con el desarrollo? * ¿Se ha establecido la frecuencia de las auditorías y los procesos necesarios para realizarlas? * ¿Se han definido políticas de seguridad y retención para los datos de artefactos generados? |

|  |
| --- |
| Temática: Responsabilidades y Colaboración |
| Definir claramente las responsabilidades y fomentar la colaboración entre equipos es esencial para asegurar la seguridad en el desarrollo y adquisición de software. Cada miembro del equipo debe conocer sus roles y responsabilidades, y estos deben estar alineados con las políticas de seguridad de la organización. Evaluar estos controles garantiza que se mantenga una estructura organizativa clara, se promueva una cultura de seguridad y se faciliten procesos de desarrollo y adquisición más seguros y eficientes.  Este levantamiento puede incluir análisis específicos tales como:   * ¿Se ha asignado claramente la responsabilidad de crear y gestionar artefactos cuando las herramientas no puedan generarlos automáticamente? * ¿Se promueve la colaboración efectiva entre los roles de seguridad y otros miembros del equipo para asegurar una integración fluida de las herramientas de seguridad? * ¿Existen planes de capacitación específicos para los roles relacionados con la utilización y gestión de herramientas en el desarrollo de software? * ¿Se actualizan estos planes de capacitación regularmente para adaptarse a cambios en el entorno de seguridad o en las herramientas utilizadas? * ¿La alta dirección ha designado un líder o equipo específico responsable del proceso de desarrollo seguro? * ¿Se comunica de manera efectiva el compromiso de la alta dirección con el desarrollo seguro a todo el personal? * ¿Se llevan a cabo campañas educativas para informar al personal sobre la importancia de la seguridad en el desarrollo de software y el uso adecuado de herramientas? |