

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

**Preguntas de Auditoría**

**Definición de Requisitos de Seguridad**

**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 definición de requisitos de seguridad es un proceso esencial para garantizar que el software desarrollado o adquirido cumpla con los estándares de seguridad de la organización y las normativas aplicables. Desde el punto de vista de la auditoría, este proceso debe ser claro, documentado y alineado con los objetivos de seguridad, permitiendo evaluar tanto el cumplimiento como las áreas de mejora. La adecuada identificación de requisitos funcionales, obligaciones externas y criterios de selección asegura un enfoque estructurado que minimice riesgos en todo el ciclo de vida del software.

**1.1 Derivación y Definición de Requisitos**

Los requisitos de seguridad deben derivarse de los requisitos funcionales y de las necesidades del cliente o la organización, garantizando que sean específicos, medibles y razonables. Estos requisitos deben estar alineados con las políticas internas y las líneas base de seguridad de la organización. Desde la perspectiva de un auditor, es importante verificar que estos requisitos sean consistentes y estén documentados adecuadamente, lo que facilita la evaluación de su cumplimiento en las etapas clave del ciclo de vida del software.

**1.2 Identificación de Obligaciones y Criterios de Selección**

Es necesario identificar y reconciliar todas las obligaciones de cumplimiento externas, incluyendo normativas, estándares y acuerdos contractuales. Esto se traduce en un conjunto de requisitos específicos que deben reflejarse en contratos, documentos de adquisición y acuerdos con terceros. Los criterios de selección del software deben incluir aspectos como la capacidad del proveedor para divulgar vulnerabilidades, responder a incidentes de seguridad y adherirse a las prácticas definidas por la organización. Un auditor revisará si estas obligaciones y criterios están claramente establecidos y si las partes involucradas los cumplen de manera consistente.

**1.3 Políticas de Diseño, Seguridad y Ciclo de Vida**

Las políticas deben especificar requisitos de diseño y arquitectura basados en análisis de riesgos, como la modularidad del código, el aislamiento de componentes de seguridad y la protección de infraestructuras de desarrollo. Estas políticas deben abarcar todo el ciclo de vida del software, desde el desarrollo hasta la notificación del fin de soporte y la eliminación segura del software. Desde el punto de vista de la auditoría, es fundamental evaluar cómo estas políticas se implementan y si se verifica su cumplimiento en puntos clave del SDLC, incluyendo pruebas automatizadas y revisiones de vulnerabilidades.

**1.4 Gestión de Relaciones con Terceros**

Los acuerdos con proveedores deben establecer responsabilidades claras y niveles de servicio relacionados con la seguridad, incluyendo la certificación de que el software cumple con los requisitos de la organización. Es importante exigir a los proveedores datos de procedencia y mecanismos de verificación de integridad para todos los componentes. Un auditor analizará si estos acuerdos son consistentes y si los procesos establecidos permiten mitigar riesgos cuando los componentes de software de terceros no cumplen con los requisitos definidos.

**1.5 Revisión, Actualización y Cumplimiento**

Los requisitos de seguridad deben revisarse al menos anualmente, o antes si surgen nuevos requisitos o incidentes importantes. Esto incluye educar a las partes interesadas sobre los cambios y establecer procesos para manejar excepciones. Además, los informes periódicos de cumplimiento deben generarse y distribuirse a todas las partes interesadas, permitiendo identificar áreas de mejora y priorizar esfuerzos de corrección. Desde una perspectiva de auditoría, estos informes son clave para evaluar el estado del cumplimiento y la efectividad de los procesos relacionados con los requisitos de seguridad.

# Modelo de Madurez

|  |  |
| --- | --- |
| NIVEL 1 [Impredecible y reactivo]: La actividad se completa, pero a menudo se retrasa y supera el presupuesto. | 1. Los equipos derivan los requisitos de seguridad de los requisitos funcionales y de las preocupaciones del cliente o de la organización.  1.1 Los requisitos de seguridad son específicos, mensurables y razonables.  1.2 Los requisitos de seguridad están en consonancia con la línea de base de la organización. 2. Ha identificado todas las fuentes de obligaciones de cumplimiento externas  2.1 Ha capturado y conciliado las obligaciones de cumplimiento de todas las fuentes. 3. Definir un conjunto básico de requisitos de seguridad para componentes de software e incluirlo en documentos de adquisición, contratos de software y otros acuerdos con terceros. 4. Definir criterios relacionados con la seguridad para seleccionar software; los criterios pueden incluir el programa de divulgación de vulnerabilidades del tercero y las capacidades de respuesta a incidentes de seguridad del producto o la adherencia del tercero a las prácticas definidas por la organización.  5. Los requisitos de seguridad se discuten con el proveedor cuando se crean los acuerdos con éste.  5.1 Los acuerdos con los proveedores proporcionan orientaciones específicas sobre la corrección de defectos de seguridad en un plazo acordado.  5.2 La organización dispone de un modelo de acuerdo de responsabilidades y niveles de servicio para procesos clave de seguridad del proveedor |
| NIVEL 2 [Gestionado a nivel de proceso]: Las actividades se planifican, ejecutan, miden y controlan. | 1. Definir políticas que especifiquen requisitos de diseño y arquitectura de software basados ​​en riesgos, como hacer que el código sea modular para facilitar la reutilización y las actualizaciones del código; aislar componentes de seguridad de otros componentes durante la ejecución; evitar comandos y configuraciones no documentados; y proporcionar características que ayudarán a los adquirentes de software con la implementación, operación y mantenimiento seguros del software.  2. Definir políticas que especifiquen los requisitos de seguridad para el software de la organización y verificar el cumplimiento en puntos clave del SDLC (por ejemplo, clases de fallas de software verificadas por puertas, respuestas a vulnerabilidades descubiertas en el software lanzado).  3. Definir políticas que especifiquen qué se debe archivar para cada versión de software (p. ej., código, archivos de paquetes, bibliotecas de terceros, documentación, inventario de datos) y durante cuánto tiempo se debe conservar según el modelo SDLC, el final del software y otros factores.  4. Definir políticas para proteger las infraestructuras de desarrollo de software y sus componentes, incluidos los puntos finales de desarrollo, en todo el SDLC y mantener esa seguridad.  5. Definir políticas para proteger los procesos de desarrollo de software en todo el SDLC y mantener esa seguridad, incluso para los componentes de software de código abierto y otros de terceros utilizados por el software que se está desarrollando.  6. Analizar el riesgo de las pilas de tecnología aplicables (por ejemplo, lenguajes, entornos, modelos de implementación) y recomendar o exigir el uso de pilas que reduzcan el riesgo en comparación con otras.  7. Garantizar que las políticas cubran todo el ciclo de vida del software, incluida la notificación a los usuarios del inminente fin del soporte del software y la fecha de fin de vida útil del software.  8. Exigir a terceros que certifiquen que su software cumple con los requisitos de seguridad de la organización.  9. Exigir a terceros que proporcionen datos de procedencia y mecanismos de verificación de integridad para todos los componentes de su software.  10. Establecer y seguir procesos para abordar el riesgo cuando existen requisitos de seguridad que los componentes de software de terceros que se adquirirán no cumplen; esto debería incluir revisiones periódicas de todas las excepciones aprobadas a los requisitos. |
| NIVEL 3 [Proactivo antes que reactivo]: Los estándares de toda la organización brindan orientación a través de proyectos, programas y carteras. | 1. Revisar y actualizar los requisitos de seguridad al menos una vez al año, o antes si hay nuevos requisitos de fuentes internas o externas, o si se ha producido un incidente de seguridad importante dirigido a la infraestructura de desarrollo de software.  1.1 Educar a las personas afectadas sobre cambios inminentes en los requisitos 2. Revisar todos los requisitos de seguridad al menos una vez al año, o antes si hay nuevos requisitos de fuentes internas o externas, se descubre una vulnerabilidad importante en el software lanzado o se ha producido un incidente de seguridad importante dirigido al software desarrollado por la organización.   2.1 Establecer y seguir procesos para manejar solicitudes de excepción de requisitos, incluidas revisiones periódicas de todas las excepciones aprobadas. 3. Dispone de procedimientos (automatizados, si es posible) para generar periódicamente informes de cumplimiento  3.1 Entrega los informes de cumplimiento a todas las partes interesadas.  3.2 Las partes interesadas utilizan la información sobre el estado de cumplimiento para identificar áreas de mejora. |
| NIVEL 4 [Gestionado Cuantitativamente]: La organización está basada en datos con objetivos cuantitativos de mejora del desempeño que son predecibles | 1. Los equipos derivan los requisitos de seguridad de los requisitos funcionales y de las preocupaciones del cliente o de la organización.  1.1 Los requisitos de seguridad son específicos, mensurables y razonables.  1.2 Los requisitos de seguridad están en consonancia con la línea de base de la organización 2. Asigne cada obligación de cumplimiento externa a un conjunto bien definido de requisitos de aplicación requisitos de la aplicación  2.1 Definir procedimientos de verificación, incluidas pruebas automatizadas, para comprobar el cumplimiento con los requisitos relacionados con el cumplimiento 3 Se miden los indicadores clave de rendimiento que tengan que ver con los requisitos de seguridad en el cumplimiento de políticas y relación con proveedores |
| 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. Las partes interesadas utilizan la información sobre el estado de cumplimiento para identificar y priorizar los esfuerzos de corrección de las deficiencias. |

1. **Preguntas de Auditoría para la Definición de Requisitos de Seguridad**

|  |
| --- |
| Temática: Definición de Requisitos de Seguridad |
| La revisión de la Definición de Requisitos de Seguridad en el Desarrollo y Adquisición de Software es fundamental para garantizar la efectividad y solidez de las medidas de seguridad implementadas en el proceso. A través de la auditoría, se busca validar que los requisitos establecidos aborden adecuadamente los riesgos y amenazas específicos, proporcionando una capa esencial de protección desde las etapas iniciales del desarrollo hasta la adquisición de software. Además, la auditoría contribuye a la identificación temprana de posibles brechas en los requisitos de seguridad, permitiendo correcciones oportunas y mejoras continuas. En resumen, auditar la Definición de Requisitos de Seguridad es clave para garantizar la robustez y la eficacia de las medidas de seguridad integradas en el software, fortaleciendo así la postura de seguridad global de la organización.  Este levantamiento puede incluir análisis específicos tales como:   * ¿Cómo se asegura el equipo de desarrollo de que los requisitos de seguridad se derivan de manera adecuada de los requisitos funcionales y de las preocupaciones del cliente u organización?   + ¿Se verifican y validan los requisitos de seguridad para asegurar que son específicos, mensurables y razonables?   + ¿Cómo se garantiza que los requisitos de seguridad estén alineados con la línea de base de la organización?   + ¿Existe un conjunto básico de requisitos de seguridad definidos para componentes de software?     - ¿Estos requisitos se incorporan de manera efectiva en documentos de adquisición, contratos de software y otros acuerdos con terceros? * ¿Cómo se definen y aplican los criterios relacionados con la seguridad para seleccionar software?   + ¿Se consideran aspectos como el programa de divulgación de vulnerabilidades y las capacidades de respuesta a incidentes de seguridad al seleccionar software de terceros?   + ¿Cómo se garantiza que los requisitos de seguridad se discuten adecuadamente con los proveedores durante la creación de acuerdos?   + ¿Existe un modelo de acuerdo que establece claramente las responsabilidades y niveles de servicio relacionados con procesos clave de seguridad del proveedor? * ¿Cómo se definen las políticas que especifican requisitos de diseño y arquitectura de software basados en riesgos?   + ¿Se consideran elementos como modularidad, aislamiento de componentes de seguridad y la evitación de comandos no documentados en estas políticas?   + ¿Existen políticas que especifiquen requisitos de seguridad para el software de la organización?   + ¿Cómo se verifica y garantiza el cumplimiento de estas políticas en puntos clave del SDLC, como clases de fallas de software verificadas por puertas y respuestas a vulnerabilidades descubiertas? * ¿Hay políticas definidas sobre qué archivar para cada versión de software y por cuánto tiempo se debe conservar?   + ¿Estas políticas se alinean con el modelo SDLC, el final de la vida del software y otros factores relevantes?   + ¿Se han definido políticas para proteger las infraestructuras de desarrollo de software y sus componentes durante todo el SDLC?   + ¿Cómo se mantiene la seguridad en estas infraestructuras y procesos a lo largo del tiempo? * ¿Se exige a terceros que certifiquen que su software cumple con los requisitos de seguridad de la organización?   + ¿Qué procesos se siguen para obtener datos de procedencia y mecanismos de verificación de integridad para componentes de software de terceros?   + ¿Existe un proceso establecido para abordar el riesgo cuando los componentes de software de terceros no cumplen con los requisitos de seguridad?   + ¿Cómo se llevan a cabo revisiones periódicas de todas las excepciones aprobadas a los requisitos? * ¿Cómo se manejan las solicitudes de excepción de requisitos y cómo se llevan a cabo revisiones periódicas de las excepciones aprobadas?   + ¿La organización dispone de procedimientos, preferiblemente automatizados, para generar informes periódicos de cumplimiento?     - ¿Estos informes se entregan a todas las partes interesadas relevantes?     - ¿Cómo utilizan las partes interesadas la información   + ¿Se identifican y abordan áreas de mejora basadas en la información obtenida de los informes de cumplimiento? * ¿Se asigna cada obligación de cumplimiento externa a un conjunto bien definido de requisitos de aplicación?   + ¿Existen procedimientos de verificación, incluyendo pruebas automatizadas, para evaluar el cumplimiento con los requisitos relacionados con el cumplimiento?   + ¿Se miden indicadores clave de rendimiento relacionados con los requisitos de seguridad y el cumplimiento de políticas? * ¿Cómo se comunica y presenta la información sobre el estado de cumplimiento a las partes interesadas?   + ¿Existen procesos establecidos para que las partes interesadas revisen y comprendan el estado de cumplimiento?   + ¿Se han identificado deficiencias en el cumplimiento de requisitos de seguridad?   + ¿Cómo se priorizan las deficiencias identificadas?   + ¿Las partes interesadas participan activamente en la identificación y priorización de esfuerzos para corregir las deficiencias?   + ¿Se realiza un seguimiento de la implementación de correcciones y se informa regularmente a las partes interesadas sobre el progreso? |