MINISTERIO DE AMBIENTE Y DESARROLLO SOSTENIBLE

Requisitos de seguridad para las buenas prácticas en el desarrollo seguro

PROCESO

Gestión Estratégica de Tecnologías de la Información

Versión 1

22/12/2022

MADSIG
Sistema Integrado de Gestión

TABLA DE CONTENIDO

LISTA DE TABLAS

1. INTRODUCCIÓN

La digitalización de los procesos y servicios tecnológicos conlleva a tener una gran responsabilidad teniendo presente que, al migrar la información a plataformas en línea, se está exponiendo la información a diferentes amenazas y delitos informáticos, por lo que es importante para las organizaciones tener una infraestructura segura de TIC que este diseñada, controlada y operada por personas que entiendan la importancia de mitigar y prevenir las amenazas también que conozcan los requisitos de seguridad y tengan las habilidades para construir y operar sistemas seguros.

El desarrollo de software seguro es un elemento fundamental dentro del contexto de seguridad de la información independientemente del lenguaje de programación que sea empleado, los cuales podrían ser susceptibles de presentar diferentes tipos de vulnerabilidades impactando en la funcionalidad de la solución tecnológica o extenderse a una falla de seguridad afectando los pilares principales de la información, disponibilidad, confidencialidad e integridad.

Lo que se pretende con este documento es establecer los lineamientos mínimos que permita gestionar el desarrollo seguro teniendo en cuenta que el diseño y desarrollo de aplicaciones debe priorizar la seguridad del ciclo de vida del software desde el inicio de su concepción, con el propósito de minimizar los riesgos que permitan detectar y solucionar de manera oportuna las fallas que se puedan presentar. Con ello a su vez permite identificar causas y evitar que se repitan errores comunes de seguridad, reduciendo la probabilidad de que se presente vulnerabilidades en el producto final.

En este lineamiento establecerá las condiciones que permitan controlar que el ciclo de vida del desarrollo de software cumpla con las consideraciones establecidas por la oficina de tecnología para de esta manera construir instrumentos que soporten el apropiado manejo de la información y uso por parte de funcionarios, colaboradores y terceros.

2. OBJETIVO

Garantizar la aplicación de las buenas prácticas en la etapa de desarrollo de sistemas de información, incluyendo aspectos de seguridad de la misma; como parte integral del proceso de desarrollo de los sistemas de información del Ministerio de Ambiente y Desarrollo Sostenible.

3. ALCANCE

Definir los lineamientos y directrices de seguridad de la información que se deben tener en cuenta en el desarrollo de las diferentes fases de la construcción de sistemas de información y desarrollo de software.

En el presente documento se enumerarán los principios de seguridad y buenas prácticas que deben ser tenidas en cuenta a la hora de desarrollar una aplicación de forma segura, con el fin de garantizar la seguridad de la información.

4. MARCO CONCEPTUAL

- **Desarrollo seguro de software:** se basa en la realización de validaciones de seguridad continua del proyecto en construcción, desde su fase inicial hasta la culminación del proyecto.[1]
- **S-SDLC (Secure Software Development Life Cycle**). Metodología que permite validar los requisitos de seguridad a lo largo de las distintas fases de construcción del software: análisis, diseño, desarrollo, pruebas y mantenimiento.
- **Desarrollador**: Programador o proveedor que se dedica a uno o más aspectos del proceso de desarrollo de software.[2]
- **Hash**: algoritmo matemático que transforma cualquier bloque arbitrario de datos en una nueva serie de caracteres con una longitud fija.[3]
- **Log**: realización del registro secuencial en un archivo o en una base de datos de todos.
- **Confidencialidad:** Según la norma ISO/IEC 27001, es la propiedad de que la información no esté disponible ni sea revelada a individuos, entidades o procesos no autorizados.
- **Disponibilidad:** Según la norma ISO/IEC 27001, es la propiedad de salvaguardar la exactitud y estado completo de los activos de información.
- **Integridad:** Según la norma ISO/IEC 27001, es la propiedad de que la información sea accesible y utilizable por solicitud de un individuo o entidad autorizada cuando se requiera.
- **Riesgo:** Toda posibilidad de ocurrencia de aquella situación que pueda entorpecer el desarrollo normal de las funciones de la empresa e impidan el logro de sus objetivos.
- **Templete**: Se considera la combinación de archivos que hacen parte de la estructura de la combinación de archivos que componen la parte visual de un website.[4]

[1] (Limited, s.f.)
[2] (Wikipedia, s.f.)
[3] (Latam, s.f.)
[4] (Hernandez, 2017)

5. MARCO NORMATIVO

El marco normativo aplicable al desarrollo seguro de software comprende:

- Política de Privacidad y Confidencialidad del de la información estipulada por el Ministerio, así como la Ley 1581 de protección de datos personales.
- Lineamientos que se encuentran en el manual de políticas específicas de Seguridad de Información del Ministerio en el **Adquisición, Desarrollo Y Mantenimiento de Sistemas de Información Desarrollo de software.**
- Lineamientos de control definidos **ISO/IEC 27002:2013** específicamente los controles Dominio **A.14 ADQUISICIÓN, DESARROLLO Y MANTENIMIENTO DE LOS SISTEMAS DE INFORMACIÓN.**
- Control **A.12.1.4[5] Separación de los ambientes para desarrollo, prueba y operación** el fin de este control es permitir un ambiente de desarrollo, operación y prueba sirva para reducir los riesgos y evitar acceso no autorizados.

6. RESPONSABILIDADES

- **Área funcional**: es el responsable y líder del proyecto del desarrollo del sistema de información, es el área donde se crea la necesidad, se encarga de proveer los recursos para la gestión del todo el proceso de desarrollo. Definen el alcancen funcional del sistema de información, realizan pruebas funcionales y son los encargados del uso de todas las funcionalidades desarrolladas en el sistema de información.
- **Oficina Tic:** Responsable de apoyar desde el punto de vista técnico todo el ciclo de vida del sistema de información, generan lineamientos para la gestión del ciclo de vida de los sistemas de información, ejecutan las pruebas no funcionales (requisitos no funcionales); proveer los ambientes de pruebas y de producción y Soportar técnicamente el sistema de información una vez que esté funcionando.
- **Empresas desarrolladoras / contratistas desarrolladores:** desarrollan todos los productos correspondientes a las fases de construcción de un sistema de información conforme a los requerimientos del área funcional y los lineamientos (procedimiento gestión de proyectos de sistema de información) definidos por la oficina TIC del Ministerio.

[5] (27001, s.f.)

7. POLITICA

El Ministerio velará por que el desarrollo interno o externo de aplicaciones o sistemas de información cumpla con los requerimientos de seguridad esperados, mediante la aplicación de las buenas prácticas para el desarrollo seguro, así como la metodología para la realización de pruebas de aceptación y seguridad del software desarrollado.

Se debe verificar que los desarrollos estén debidamente documentados, así como todas las versiones del desarrollo se deben preservar adecuadamente en varios medios y guardar una copia de respaldo en sitio externo.

8. PRINCIPIOS BÁSICOS

Este documento se basa en los 10 principios de desarrollo seguro:

- Partir siempre de un modelo de permisos mínimos, es mejor ir escalando privilegios por demanda de acuerdo a los perfiles establecidos en las etapas de diseño.
- Si se utiliza un lenguaje que no sea compilado, asegurarse de limpiar el código que se pone en producción, para que no contenga rutinas de pruebas, comentarios o cualquier tipo de mecanismo que pueda dar lugar a un acceso indebido.
- Nunca confiar en los datos que ingresan a la aplicación, todo debe ser verificado para garantizar que lo que está ingresando a los sistemas es lo esperado y además evitar inyecciones de código.
- Hacer un seguimiento de las tecnologías utilizadas para el desarrollo. Estas van evolucionando y cualquier mejora que se haga puede dejar obsoleta o inseguras versiones anteriores.
- Todos los accesos que se hagan a los sistemas deben ser validados.
- Para intercambiar información sensible utilizar protocolos para cifrar las comunicaciones, y en el caso de almacenamiento la información confidencial debería estar cifrada utilizando algoritmos fuertes y claves robustas.
- Cualquier funcionalidad, campo, botón o menú nuevo debe agregarse de acuerdo a los requerimientos de diseño. De esta forma se evita tener porciones de código que resultan siendo innecesarias.
- La información almacenada en dispositivos móviles debería ser la mínima, y más si se trata de contraseñas o datos de sesión. Este tipo de dispositivos son los más propensos a ser que se pierdan y por lo tanto su información puede ser expuestas más fácilmente.
- Cualquier cambio que se haga debería quedar documentado, esto facilitará modificaciones futuras.
- Poner más cuidado en los puntos más vulnerables, no hay que olvidar que el nivel máximo de seguridad viene dado por el punto más débil. (welivesecurity, 2014)

9. ASPECTOS GENERALES

Las siguientes consideraciones deberán ser concebidas inicialmente para el desarrollo de software, donde se incorporar las buenas prácticas de seguridad a lo largo de todas fases de la metodología del desarrollo del software.

9.1. Requisitos de Seguridad

a) Análisis de Requerimientos

Esta etapa de deberán considerar los siguientes aspectos:

- Considerar los requerimientos funcionales y no funcionales.
- Revisar los requerimientos desde el punto de vista de ejecución en el tiempo proyectado.
- Ciclos de desarrollo estandarizado que incluyan los criterios de seguridad y calidad.
- Los requerimientos deben ser verificables.
- Identificar los aspectos claves de seguridad y control durante las fases de desarrollo del proceso; al igual que el procedimiento de evaluación de riesgos.
- Esta evaluación deberá estar documentada reflejando los requisitos solicitados para el desarrollo y arquitectura de la aplicación como de acuerdo al handover definido para el proyecto y con los lineamientos de la Oficina TIC de Min Ambiente, por ejemplo:
    - Tipo de arquitectura.
    - Plataforma donde correrá la aplicación.
    - Seguridad.
    - Tipo de datos almacenar
    - Requerimientos normativos y marcos regulatorios.
    - Tipos de registro que la aplicación debe generar como modo de acceso; privilegios que se van asignar a los usuarios.
    - Creación de perfiles los cuales deben estar definidos de acuerdo a lo solicitado por las partes interesadas definiendo las acciones que puede realizar cada perfil (lectura, escritura, modificación y eliminación).
    - La aplicación deberá contar con mecanismos de autenticación para su ingreso con el fin de garantizar la integridad de la misma.
    - Se deben realizar acciones preventivas y correctivas durante el proceso de desarrollo y prueba del mismo.

- En caso de realizar cambios estos deberán ser documentados donde se evidencie en el formato **F-A-GTI-001 Formato control de cambios** se debe verificar si estos son posibles o, se rechaza o se pospone. Con la finalidad de que todos los cambios propuestos sean tratados de forma consistente y controlada.

- Determinar y documentar los permisos para acceder a los activos de información por parte de los desarrolladores. Específicamente aquella que sea considerada como sensible.

a) Arquitectura y Diseño

Manejar una arquitectura mínima de tres capas.

En esta fase se define la arquitectura y diseño donde se determina la arquitectura y estructura del sistema de software. Además, proporciona en algunos casos la base de la lógica para codificar de tal manera que se cumpla con la especificación de los requerimientos.
Se debe tener en cuenta los siguientes aspectos:

- El diseño debe ser flexible permitiendo con facilidad aplicar futuros cambios en sus requerimientos.
- Este debe facilitar la adición de nuevas funcionalidades.
- El diseño debe ser eficiente fácil de comprender e implementar.
- Debe ser seguro evidenciando: autorización, autenticación, mensajes de error, Mecanismos de protección de datos; Codificación de software ya que esto evita vulnerabilidades.
- Pruebas de software el objetivo es encontrar y reportar errores funcionales de la aplicación desarrollada.

9.1.1. Fase de desarrollo

Tanto el área encargada como los responsables del desarrollo deberán seguir unas reglas o estándares, las cuales ayudaran a mantener el código legible y fácil de comprender algunas de ellas son:

- Comentar el código lo cual facilitara su comprensión: El código que se desarrolle en el marco de los sistemas de información del ministerio debe estar identificado y en lo posible cada sección de código debe contar con sus respectivos comentarios donde

se evidencia su propósito. Y esto ayude a que el código sea fácil de entender y reflejen porque el código está haciendo lo que está haciendo.

- Nombrar las variables de una forma que sea entendible de acuerdo a las funciones que realiza tanto en la interfaz como en el código: El código debe ser escrito de forma que sea fácil de leer por otros desarrolladores; que sea fácil de entenderlo con poco tiempo y esfuerzo.
- Mantener el código limpio y organizado.
- Siempre dividir el proyecto por paquetes para entender a donde corresponde cada interfaz, clases e imágenes.
- El código debe ser fácil de mantener.
- Evitar duplicaciones de código.
- Utilizar las características del lenguaje de programación para el desarrollo de un código óptimo que permita un mejor desempeño del sistema de información.
- Las clases de alto nivel no deben tener dependencias con clases de bajo nivel.
- No implementar funcionales en el código que no sean necesarias.
- Dejar el código limpio cada vez que se termine y pruebe alguna funcionalidad.
- Utilizar los principios Solidos son las buenas prácticas para ayudar a escribir un mejor código: más limpio, sostenible y escalable.
- El código fuente debe ser alojado y versionado en el repositorio asignado en la oficina TIC de Minambiente (Guía de manejo de repositorio de código fuente).
- Se deberá utilizar el témplate (diseño web que tiene la entidad) que la oficina de tecnología asigne para el desarrollo de acuerdo a los lenguajes base que tiene definidos la entidad.
- Se debe utilizar un corrector de código a medida que se va escribiendo el código. (dicho corrector será sugerido por la oficina de tecnología).
- Una vez la empresa desarrolle el código fuente, deberá garantizar la calidad del código fuente, realizando validaciones y de revisiones de calidad sobre el mismo

antes de entregarlo al Ministerio, esto con el fin de asegurar el buen funcionamiento técnico del desarrollo.

- Se debe definir conjuntamente con la oficina TIC de Minambiente el porcentaje de pruebas unitarias y cypress el cual permite comprobar que la performance de un producto de software recién desarrollado sea buena y corresponda con los requerimientos iniciales para garantizar su adecuado funcionamiento.
- Dichas pruebas deben ser ejecutadas por el proveedor con la participación de los delegados que asigne la oficina de tecnología del Ministerio al igual que las pruebas de funcionalidad se deberán hacer el área designada por ellos esto con el fin de garantizar que el momento de correr la aplicación funcione en el ambiente asignado por el Ministerio.

9.1.2. Procedimiento para Control de Cambios en los Sistemas de Información.

Para cualquier cambio en el código, los controles de cambios, así como el proceso de cambio se deben aplicar para asegurar la integridad del sistema conforme a lo establecido en el SGSI del Ministerio de Ambiente y Desarrollo Sostenible. Además del cambio de controles ya necesarios, también se deben considerar los siguientes controles específicos listados a continuación:

Controles para la Gestión de Control de Cambios en los Procesos de Desarrollo
Garantizar la actualización de la documentación del sistema después de la finalización de cada cambio
Mantener un control de versiones para todas las actualizaciones de software
Realizar revisiones siempre que sea necesario, para actualizar su contenido o se cumplan los plazos máximos establecidos para ello.
Asegurar que se actualizará toda la documentación y los manuales o procedimientos operativos de los usuarios

Tabla 1 controles Gestión de Cambio

Los cambios deben ser realizados de acuerdo a los lineamientos definidos por la oficina TIC de Minambiente para el manejo de repositorio (repositorio de Código)

10. VALIDACIÓN DE SOFTWARE

En esta etapa las partes interesadas realizarán un conjunto de procesos de análisis y comprobación asegurando que el software que se desarrolla vaya acorde a su especificación y cumple las necesidades solicitadas por el Ministerio.

- Dentro de los pasos a seguir se validarán siempre los datos de entrada antes de procesarlos esto con el fin de comprobar que los datos con correctos; para evitar que personas ajenas o malintencionadas tenga acceso a la información.
- Se deberá controlar el tamaño y tipos de datos de entrada de esta forma se evitará el ingreso de información no autorizada y que pueda poner en riesgo la aplicación.
- Se eliminará caracteres especiales esto evitara que no se puedan ingresar sentencias de programación que pudieran dar lugar a procesarlas por la aplicación.
- Transformar los datos de entrada en una codificación establecida por el área desarrolladora y el Ministerio, lo que se busca es que el código quede uniforme y los datos no se mezclen con el mismo.
- Se debe realizar un análisis de código tanto estático como dinámico a fin de garantizar que, al emplear nombres descriptivos, en la declaración de variables, es decir, que hagan alusión a su nombre y no a su tipo.
- Se deben capturar errores de capas inferiores y no mostrarlos al usuario

En lo referente a la creación y asignación de contraseñas se tendrán en cuenta los siguientes lineamientos:

- Deberán contar con una longitud mínima de caracteres de acuerdo a la política establecida por el área de seguridad.
- No podrá incluir nombres y apellidos del usuario, así como fechas evidentes ya sea año de nacimiento mes de nacimiento o cualquier tipo de información que sea relevante del usuario.
- La contraseña deberá ser modificada por lo menos una vez semestralmente por el usuario de la cuenta, esta será modificada por primera vez cuando le sea asignado al usuario.

11. MANTENIMIENTO Y EVALUACIÓN DE SOFTWARE

Esta fase involucra cambios al software para corregir defectos encontrados durante su uso y prueba, así como la adición de nueva funcionalidad para mejorar la usabilidad y aplicabilidad del software.

Los tipos de mantenimiento que se tomaran como guía durante y uso del desarrollo son:

- **Mantenimiento evolutivo:** Mejora del software (rendimiento, flexibilidad, reusabilidad) o implementación de nuevos requisitos. También se conoce como mantenimiento evolutivo.
- **Mantenimiento adaptativo:** se validará la adaptación del software a cambios en su entorno tecnológico en dado caso la implementación de hardware, una nueva base de datos o un nuevo sistema operativo garantizando que en el momento de que esto llegará a suceder el aplicativo se va adaptar de tal forma que no pueda perjudicar la operación.
- **Mantenimiento correctivo:** en este se tiene previstas aquellas correcciones a los fallos detectados durante la explotación.
- **Mantenimiento preventivo:** se validará el desarrollo con el fin de facilitar el mantenimiento futuro del sistema (verificar precondiciones, mejorar legibilidad).

El Ministerio de Ambiente deberá aplicar pruebas de verificación de la seguridad en cada componente y sus interacciones. La responsabilidad de estas pruebas pueda recaer en la responsabilidad del desarrollador, o bien del responsable de verificación de las mismas.

Revisión de Código	Verificar que el código fue desarrollado de acuerdo con las normas de desarrollo que se describen en este documento. Un segundo desarrollador (podrá ser contratista o tercero) y el autor del código fuente, deben realizar revisiones del código en cada software nuevo o modificado, sobre todo en tratar de identificar problemas de seguridad. Quien revise el código debe tener los conocimientos necesarios sobre el proceso de revisión de código y prácticas de desarrollo seguro. Todos los parches deben ser desplegados y los resultados de la revisión de código deben ser revisados y aprobados por la oficina TIC antes de su eliminación.
Control de Calidad	La aplicación de un control de calidad no debe comprometer los controles de seguridad existentes, o introducir nuevas vulnerabilidades.
Pruebas Funcionales y de Seguridad de Calidad	Además de las pruebas de eficiencia y funcionalidades, todas las funciones de seguridad de aplicación también deben ser probadas.
Documentación	Todas las funciones de la aplicación y la documentación acerca de su implementación deben incluir instrucciones sobre su configuración segura.
Paso a Producción	La implementación de la aplicación en el ambiente de producción no debe comprometer los controles de seguridad existentes, o introducir nuevas vulnerabilidades.

Revisión de Código	Verificar que el código fue desarrollado de acuerdo con las normas de desarrollo que se describen en este documento. Un segundo desarrollador (podrá ser contratista o tercero) y el autor del código fuente, deben realizar revisiones del código en cada software nuevo o modificado, sobre todo en tratar de identificar problemas de seguridad. Quien revise el código debe tener los conocimientos necesarios sobre el proceso de revisión de código y prácticas de desarrollo seguro. Todos los parches deben ser desplegados y los resultados de la revisión de código deben ser revisados y aprobados por la oficina TIC antes de su eliminación.
Pruebas en Producción	Además de las pruebas y de eficiencia y funcionalidad, todas las funciones de seguridad de la aplicación deben ser probadas.
Mantenimiento	Todos los mantenimientos futuros de la aplicación no deben comprometer los controles de seguridad existentes, o introducir nuevas vulnerabilidades. Cualquier nuevo código debe seguir el mismo flujo del ciclo de vida. (OWASP, s.f.)

Tabla 2 Ciclo de Desarrollo en el Ministerio de Ambiente y Desarrollo Sostenible.

Todos los desarrolladores del Ministerio de Ambiente y Desarrollo Sostenible, así como terceros o contratistas asociados en este proceso o servicio, deben estar capacitados en prácticas de desarrollo de seguro. El desarrollo interno o subcontratado de aplicaciones propietarias deben utilizar técnicas de desarrollo seguro reconocidos por la industria para prevenir vulnerabilidades conocidas.

a. Todos los desarrolladores en el Ministerio de Ambiente y Desarrollo Sostenible deben considerar al menos las siguientes vulnerabilidades durante la revisión de código y fases de prueba para todas las aplicaciones:

Se toma como referencia el top **10 de vulnerabilidades de OWASP**

Vulnerabilidades a ser validadas durante las fases de revisión de código y pruebas	
Inyección de Código	La inyección de código ocurre cuando los datos suministrados por el usuario son enviados a un intérprete como resultado de la ejecución de un comando o una consulta.
Buffer Overflow	El desbordamiento de búfer (buffer overflow) se produce cuando una aplicación no tiene un chequeo de límites adecuados en su espacio de búfer.

Vulnerabilidades a ser validadas durante las fases de revisión de código y pruebas	
Almacenamiento Criptográfico Inseguro	Al no utilizar recursos de cifrado fuerte adecuadamente para almacenar datos están en un mayor riesgo de verse comprometidos en exposición de las credenciales de autenticación y/o datos.
Comunicaciones Inseguras	se debe cifrar la información para evitar que el Atacante puede explotar los procesos de cifrado.
Manejo Inadecuado de Errores	Se deben utilizar mensajes de error genéricos, como "datos no pudieron verificarse." Para evitar que el atacante obtenga información fácil de validar “como contraseña incorrecta “
Vulnerabilidades Altas	Todas las vulnerabilidades identificadas por el proceso de calificación de riesgo de vulnerabilidad del Ministerio de Ambiente y Desarrollo sostenible. Como de "alto riesgo" y que podría afectar a la aplicación deben ser identificados y resueltos durante el desarrollo de aplicaciones.
Cross-site scripting (XSS)	Fallas XSS ocurren siempre que la aplicación recoge los datos facilitados por el usuario y los envía a un navegador sin primero validar o codificar ese contenido.
Control de Acceso inadecuado	Una referencia de objeto directo se produce cuando un desarrollador expone una referencia a un objeto de implementación interna, tales como archivo, directorio, registro de base de datos, o la clave, como un parámetro de URL o forma.
Cross-site request forgery (CSRF)	Un atacante de CSRF puede forzar el navegador de la víctima conectado para enviar una solicitud de pre-autenticado para una aplicación web vulnerable, que a su vez permite a un atacante realizar cualquier operación de cambio de estado a las cuales las víctimas tienen permiso para realizarla (por ejemplo, la actualización de datos de la cuenta, realizar adquisiciones o incluso autenticarse en la aplicación).
Boken authentication	Una autenticación segura y gestión de sesión impide que personas no autorizadas comprometan las credenciales, llaves o tokens de sesión legítimos de la cuenta, lo que puede permitir al atacante asumir la identidad de un usuario autorizado. (OWASP, s.f.)

Tabla 3 Vulnerabilidades a ser validadas durante las fases de revisión de código y pruebas

a. Anualmente, y siempre que se produzcan cambios significativos, todas las aplicaciones basadas en la Web deben ser sometidas a un test de penetración a nivel de la aplicación y su infraestructura.

b. Con el fin de proteger a todas las aplicaciones web contra ataques basados en la Web, por lo menos uno de los siguientes métodos debe utilizarse para nuevas aplicaciones:

- Por lo menos una vez al año o después de cualquier cambio significativo; todo el código personalizado para las aplicaciones basadas en la web debe ser revisado por área encargada (interno o externo) en seguridad a nivel de aplicación y se debe garantizar que esté separado del ambiente administrativo de las aplicaciones.
- Una vez revisado el código se deben corregir las vulnerabilidades y pueden ser utilizadas herramientas automatizadas o controles de seguridad o métodos manuales.
- Utilice un firewall de aplicaciones Web (WAF) para detectar y prevenir los ataques basados en la web.

12. AMBIENTE DE DESARROLLO SEGURO

Un entorno de prueba / desarrollo, separado del entorno de producción, debe ser utilizado para probar todo nuevo software (incluyendo parches). Si los ambientes tienen conectividad entre sí los controles de acceso deben estar disponibles para reforzar la separación o aislamiento.

a. Debe respetarse la separación de responsabilidades entre el personal asignado para entornos de prueba / desarrollo y los asignados al entorno de producción.

b. Información de producción no se utilizará con fines de prueba y desarrollo. El personal de pruebas y desarrollo deben usar sólo datos falsos (puestos a disposición por los líderes de procesos) en sistemas y pruebas de software no en producción.

c. Toda la información y las cuentas de prueba deben ser removidos antes de que el sistema o aplicación entre en producción. Además, todas las cuentas de aplicaciones personalizadas, nombres de usuario y las contraseñas deben ser removidos antes de su implementación en producción o su liberación a los usuarios finales.

d. Cada ejecución de código en el entorno de producción debe ser ejecutado por el administrador del sistema. Bajo ninguna circunstancia el Área de Desarrollo tendrán acceso de lectura y/o

escritura a todas las aplicaciones o datos de producción. En situaciones de emergencia los desarrolladores pueden ayudar en la resolución de problemas utilizando un ID de Emergencia.

12.1. Disposiciones de buenas prácticas para desarrollos realizados por terceros y/o contratistas

a. Para los desarrollos a cargo de terceros o subcontratados a una entidad externa por el Ministerio de Ambiente y Desarrollo sostenible se debe establecer un programa de seguimiento al tercero a fin de garantizar como mínimo el cumplimiento de los siguientes requisitos:

Controles para Desarrollo de Software Externo o Tercerizado
Acuerdos de licencia, propiedad del código y de los derechos de propiedad intelectual
Requisitos contractuales validando que los códigos siguen prácticas seguras de codificación, incluidas las pruebas de seguridad
Proporcionar evidencia de que los principios de seguridad se han utilizado en la codificación
Proporcionar evidencia de que se realizaron pruebas suficientes para proteger contra el código o contenido malicioso
Proporcionar evidencia de que se realizaron pruebas suficientes para proteger contra vulnerabilidades conocidas
Derechos contractuales para auditar los controles y el desarrollo de procesos

Tabla 4 Controles para Desarrollo de Software Externo o Tercerizado

Dentro de los requisitos y derechos contractuales se deberá tener en cuenta:

a) Las partes interesadas definirán los términos y las definiciones en las cuales se realizará en contrato con el fin de manejar todo en los mismos términos y a futuro no se preste mal interpretaciones en las cláusulas y objetos del contrato.

b) Se definirán las funcionalidades de la aplicación por lo que es recomendable que dentro del contrato se creen clausulas o un anexo en donde se especifiquen y se describan las características del desarrollo solicitado; algunos detalles pueden ser: de funcionamiento; características de accesibilidad, diseño, usabilidad, prueba y seguridad.

c) Se definirá dentro de los requisitos los Procedimientos que se deberán utilizar para la revisión y aprobación de proyectos, así como el tiempo de entrega; dentro de este se deberá especificar si se hará por fases o etapas siempre teniendo en cuenta el alcance del proyecto.

d) las partes interesadas pactaran realizar y efectuar el seguimiento y control del estado y porcentaje de avance del proyecto realizando reuniones que se pactaran con anticipación para determinar si serán semanales, periódicas o como ellos determinen que es conveniente y permitan realizar una mejor gestión y control de calidad de los productos que se están desarrollando; para tal fin se realizaran mesas de trabajo.

En estas mesas de trabajo se tendrán como guía los siguientes ítems:

- Cronogramas de trabajo; el cual fue pactado desde el inicio del contrato
- Los Recursos con los que se contaran y se llegaran a necesitar
- Seguimiento y aplicación de los controles de cambio
- Tipos de Riesgos a los que puede estar expuesto el proyecto
- El estado del proyecto a los referentes a lo pactado en la etapa contractual
- Evidencias que validen el cumplimiento de acuerdos de niveles de servicio.

e) Dentro de los términos y requerimiento contractuales las partes interesadas están en disposición de dejar dentro del contrato cláusulas de condiciones especiales: como que todo desarrollo es propiedad del Minambiente a menos que dentro de las obligaciones especiales se especifique que se respetara la propiedad intelectual del desarrollador.

f) Dentro de los requerimientos se debe pactar que el contratista ofrecerá soporte, acompañamiento y capacitación referente al manejo, manipulación, actualización, reinstalación de nuevas versiones y administración del producto final ya funcionando en la plataforma asignada por el Ministerio de Ambiente.

Estos se comprometerán a realiza soporte técnico en caso de ser necesario al igual que capacitarán a las personas encargadas de administrar el sistema para garantizar una óptima manipulación y funcionamiento del producto final de desarrollo.

g) Se deberá especificar cláusulas donde se pacte el periodo de garantía del producto en la cual se especificara el tiempo que establezcan o acuerden las partes, en esta garantía se reflejara el plazo en el que se pondrá en funcionamiento la aplicación y se podrán valorar las problemas de funcionamiento posibles brechas de seguridad y controles aplicar; también se especificara periodo de garantía así como la calidad de los bienes o servicios entregados y por último se dejara como evidencia acta de finalización del contrato.

h) El Contratista se comprometerá a no realizar actividades o programas similares a los solicitados por el ministerio, así como se deberá firmar una cláusula de confidencialidad que garantice la reserva y discreción en el cumplimiento de las actividades pactadas en el contrato; en caso de que se evidencie incumplimiento por parte del contratista dentro del contrató se pactara una penalidad la cual se fijara en el contrato organizado por las partes acordadas.

i) El Contratista se comprometerá a garantizar que el desarrollo creado es original y no la reutilización de otros códigos donde al incumplir esto será el mismo quien asumirá la responsabilidad, daños o perjuicios ocasionados por la vulneración de derechos de propiedad intelectual de terceros.

j) Las partes interesadas firmaran acuerdos de confidencialidad y protección de datos

k) Se fijarán cláusulas de penalidades por incumplimiento donde se refleje la cuantía estas estarán vinculadas a temas como el cumplimiento de las cláusulas de confidencialidad, protección de datos, propiedad intelectual y no competencia, entre otras.

13. GESTION DE RIESGOS

El riesgo es la probabilidad que ocurra una pérdida de datos cuando se trata de riesgos técnicos un ejemplo de esto es por mal asignación de privilegios a un usuario que queda con permisos de administrador permitiéndole modificar o borrar información del software; estos se convierten en medida de la probabilidad de que se produzcan efectos adversos en el desarrollo, mantenimiento etc.

Este lineamiento establece no solo los criterios de validación de los riesgos en el ciclo de vida del desarrollo si no a la aplicación de los controles para prevenir y mitigar futuras amenazas.

El área encargada deberá tener en cuenta las siguientes acciones al momento de validar los riesgos en el desarrollo de aplicaciones:

- Se realizará un análisis de riesgos validando la aplicación e identificando sus componentes con el fin de determinar las amenazas.
- Determinar tipo de acciones y técnicas y tecnologías necesarias para mitigar los riesgos identificados.

Este análisis de riesgo debe documentarse formalmente y se debe considerar como obligatorio, su gestión será realizada por la oficina TIC y donde aplique bajo el acompañamiento de contratistas vinculados al desarrollo de sistemas de información cumpliendo los siguientes criterios:

Criterios de Validación en la evaluación de riesgos
Los procesos de autenticación de usuario siempre deben soportar canales seguros para proteger la identidad del usuario. Los métodos o mecanismos autorizados de Cifrado en los canales de transmisión deben ser aprobados de acuerdo con los lineamientos de criptografía establecidos por la entidad.
Procedimientos de autorización y aprovisionamiento de acceso, deben ser soportados por el sistema de información para asegurar niveles de acceso para funciones comunes y funciones privilegiadas. Además, que sólo los usuarios autenticados y los usuarios autorizados tengan acceso.
El almacenamiento de información considerada sensible y confidencial debe ser protegido durante su transmisión y almacenamiento utilizando un método seguro.
Los registros de auditoría deben existir a fin de registrar todas las acciones realizadas por los usuarios comunes y/o privilegiados, sobre todo cuando los datos sensibles y confidenciales son procesados. Los registros deben gozar de no repudio. (Cuenca, s.f.)
Todos los incluidos en el numeral 4. Principios básicos incluido en este documento

Tabla 5 Criterios para la evaluación de Riesgos

Para adquisición y desarrollo de Software funcionarios, contratistas y terceros tomaran como guía los lineamientos que se encuentran contemplados en la manual de Políticas específicas de Seguridad de Minambiente en el numeral **Adquisición, Desarrollo y Mantenimiento de Sistemas de Información Desarrollo de software".**

Como resultado de la evaluación del riesgo, los riesgos clasificados como medio, alto o crítico debe ser remediados o resueltos, o en su defecto haber aplicado controles de compensación antes de que el sistema de información entre en el ambiente de producción.

Estos lineamientos para el sistema adquirido de terceros, estos deben cumplir con las cláusulas de confidencialidad pactadas en los contratos; cumplir con los acuerdos de niveles de servicio estipulados; dar buen uso a la información de propiedad del ministerio, respetando siempre la propiedad intelectual de la misma y que sea usada para el fin al cual se destinó dicha información.

Dentro del contrato se definirán responsables y adicionara un cronograma de actividades donde se establecen compromisos y actividades de los intervinientes del proyecto.

- Los colaboradores, proveedores y contratistas deberán solicitar permiso para la instalación de software se deberá validar que este cuente con licenciamiento apropiado y acorde con la propiedad intelectual. Se debe tener en cuenta que todo licenciamiento debe ser aprobado por la oficina TIC de Min Ambiente y debe quedar registrado a nombre de la entidad.
- Las áreas encargadas del software que se encuentre en desarrollo deben participar desde el inicio hasta la terminación de todo el proceso. Con el fin de validar ajustes y cambios necesarios.
- La Oficina encargada debe verificar que el software adquirido o desarrollado pueda integrarse con los sistemas de información existentes.
- El desarrollo de software realizado por los funcionarios, contratistas o terceros en ejecución de sus obligaciones será de propiedad intelectual del Ministerios a menos que en el alcance de su contrato se especifique lo contrario se respetará la propiedad intelectual, al igual que la licencia de uso.

Toda modificación de software bien sea por actualizaciones o algún tipo de modificación, deberá ser analizada previamente en ambientes independientes de desarrollo y prueba, con el objetivo de identificar y analizar los riesgos de seguridad.

14. BUENAS PRACTICAS DE SEGURIDAD EN EL DESARROLLO

Estas recomendaciones serán aplicadas en todas las fases del desarrollo por parte de las áreas participantes en el proceso de desarrollo:

- Se deben validar que los controles aplicados en el proyecto cumplan con la seguridad que se necesita.
- Las áreas *encargadas* realizaran una inspección de código en la fase de desarrollo validando que cumpla con los requerimientos funcionales (área Temática) y no funcionales (TIC) asignados y que estos cuenten con lineamientos de seguridad especificados previamente entre las áreas encargadas.
- Se realizará una comprobación de la gestión de las configuraciones para garantizar que el sistema funcione a medida que se van realizando cambios.

- Se deberá validar todos los parámetros de programación de la aplicación (API) verificando que los datos ingresados sean válidos y coherentes; también si ha de ser necesario sea aplicaran herramientas criptográficas para proteger la información, a igual de que si es necesario deberán adquirir (API) criptográficas. La Finalidad de estas herramientas es garantizar que las comunicaciones y aplicaciones sean seguras.
- Se deben aplicar medidas preventivas que mitiguen y aseguren la aplicación de riesgos. Pruebas unitarias automatizar las pruebas y correrlas en los dos ambientes que tiene el ministerio pruebas y producción.
- Los funcionarios no podrán acceder a las aplicaciones si previa autenticación y menos atreves de redes públicas, esto solo se podrá realizar por medio de una (VPN) asignada por el área encargada del Ministerio con el objetivo de proteger la puerta de entrada de la nube.
- Referente a la utilización **JWT** este estándar nos permite la creación de Tokens de acceso para la validación de identidad y el perfil que deberá tener del usuario ya que este sirve de verificación de contenido es importante que este token evidencie el manejo de la firma digital, se recomienda (JWS, RFC 7515); la cual se puede generar usando claves simétricas de tipo (HMAC) o claves asimétricas (RSA o ECDSA).
  Adicionalmente si se considera necesario los JWT pueden contener también datos cifrados (JWE, RFC 7516) para proteger datos sensibles.
- Tanto el desarrollador como el área encargada de supervisar deberán usar algoritmos criptográficos fuertes que garanticen la integridad de la firma; se validara que este cuente con una fecha de expiración y un identificador único.
- Se tendrá que dar un valor al emisor y al emisario lo validará que se puedan garantizar la identificación de la clave asignada y verificar que efectivamente está siendo utilizada por los mismos.
- Se validará que los tokens que contengan información sensible estén debidamente cifrados con el fin de garantizar la integridad de la información ahí contenida.
- Para garantizar la integridad de la información no se aceptarán token sin firmar así que aquellos donde no se pueda validar el emisor y destinatario será eliminados.
- Se recomienda utilizar certificado de seguridad emitidos por la autoridad certificadora (CA) esto con el fin de poder habilitar el protocolo HTTPS el cual facilita la verificación de la dirección WEB validando que efectivamente esta esté vinculada a la entidad.
- Se debe aplicar un nivel alto de seguridad lo aconsejable es utilizar una clave de 2048 bits, validar que el certificado a utilizar sea confiable u ofrezca asistencia técnica; al igual que se debe validar que tipo de certificado se necesita Como, por ejemplo:
  - Un certificado único para cubrir un único dominio seguro.
  - Un certificado para varios dominios si tienes varios sitios seguros conocidos.

- Un certificado comodín si se trata de un sitio seguro con muchos subdominios dinámicos.
- Utiliza redirecciones de servidor permanentes.
- Los encargados de realizar el desarrollo, así mismo como el de supervisarlos validaran que el certificado este actualizado y vigente con el fin de garantizar la integridad y seguridad del mismo.

- Se recomienda la utilización de plataformas de licencia de código abierto que permitan intercambiar la información de forma segura a través de Internet algunas de estas pueden ser X-Road es una solución de capa de intercambio de datos de código abierto y gratuita la cual garantiza la confidencialidad, integridad e interoperabilidad entre las partes que intercambian datos permitiendo tener control sobre gestión de direcciones , Enrutamiento de mensajes, administración de derechos de acceso ,Autenticación de usuarios a nivel de organización, así como como también la Autenticación a nivel de máquina, Cifrado a nivel de transporte, la aplicación de Firma digital de mensajes; control de inicio de sesión y mensaje de manejo de errores.

14.1. Desarrollo En La Nube

Lo que concierne a la seguridad del desarrollo que es alojando en la Nube -On Premise se deberá tomar las siguientes medidas de seguridad:

- Se deberá cifrar las comunicaciones con la totalidad en la nube es decir que toda la información allí contenida puede y debe garantizar su integridad.
- Se aplicará un cifrado de extremo la ventaja de la aplicación de este es las comunicaciones no quedan a disposición de extraños ya que para poder acceder a esta se necesita una clave cifrada.
- Para la configuración de estas aplicaciones se deberá tener en cuenta algunos principios como la modificación de los ajustes predeterminados ya que estos por defecto les dan acceso a personas mal intencionadas a la puerta principal.
- No se debe dejar depósitos de almacenamiento abiertos en la nube ya que esto facilita con la URL ver el depósito de almacenamiento.
- Aparte de aplicar los lineamientos de contraseñas seguras es importante la implementación de un administrador de contraseñas esta se encargará de la administración de contraseñas.
- Realizar mantenimiento y utilización de herramientas de seguridad como firewall garantizando que su configuración y licencias siempre estén actualizadas.

- Realizar copias de seguridad de los datos periódicamente al igual que la modificación y validación de permisos.
- Si se tiene contratado este servicio las áreas encargadas se asegurarán que el proveedor cuente la suficiente Seguridad de la nube garantizando que realiza auditorias de seguridad, que este segmenta los datos, aplica cifrado y administra los accesos a sus aplicaciones garantizado la integridad, confiabilidad y disponibilidad en la nube.
- El proveedor garantizara sin ningún inconveniente la integración de las aplicaciones con los sistemas de software con los que cuente el Ministerio de Ambiente y Desarrollo Sostenible; deberá supervisar la API evitando el uso indebido; así como validar que la conexión cuente con suficiente rendimiento para evitar cuellos de botella y afectar el rendimiento del sistema cuando esté presente alto tráfico.
- El intercambio de recursos de origen cruzado (CORS) deberá garantizar la restricción de solicitudes HTTP de origen cruzado.
- Se deberá especificar el código de estado HTTP.
- Se debe definir una expresión regular para seleccionar la salida del backend reflejando la respuesta de integración.
- En caso de ser necesario el encargado deberá declarar mapeos compuestos de pares de clave-valor para asignar los parámetros de la respuesta de integración especificados a los parámetros de una respuesta de método determinada.

BIBLIOGRAFÍA

27001, I. (s.f.). *ISO27001.4-adquisicion-desarrollo-y-mantenimiento-de-los-sistemas-de-informacion*. Obtenido de https://normaiso27001.es/a14-adquisicion-desarrollo-y-mantenimiento-de-los-sistemas-de-informacion/

Cuenca, D. ,. (s.f.). Obtenido de https://owasp.org/www-pdf-archive/Desarrollo_Seguro_Principios_y_Buenas_Pr%C3%A1cticas..pdf

Hernandez, V. (11 de 05 de 2017). *C/C++: plantillas (templates) en C++*. Obtenido de https://codingornot.com/cc-plantillas-templates-en-c#:~:text=Una%20plantilla%20es%20una%20manera,cada%20versi%C3%B3n%20de%20la%20funci%C3%B3n.

ISO, T. (s.f.). *Control A.12.1.4 ISO 27001:2013 Separación de entornos de desarrollo, prueba y operacionales* . Obtenido de https://normaiso27001.es/a12-seguridad-de-las-operaciones/

Latam, K. (s.f.). *¿Qué Es Un Hash Y Cómo Funciona?* Obtenido de https://latam.kaspersky.com/blog/que-es-un-hash-y-como-funciona/2806/

Limited, S. (s.f.). *Ciclo de vida de desarrollo de software seguro (SSDLC)*. Obtenido de https://snyk.io/learn/secure-sdlc/

NACIÓN, E. C. (02 de 07 de 2021). *gov.co.* Obtenido de https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=6279

OWASP, O. (s.f.). *Los diez riesgos más críticos en Aplicaciones Web*. Obtenido de https://wiki.owasp.org/images/5/5e/OWASP-Top-10-2017-es.pdf

REPUBLICA, C. D. (03 de 2014). *Ley 1712 de 2014.* Obtenido de https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=56882

Torres, O. P. (21 de 05 de 2021). *Gestión de riesgos en proyectos de software*. Obtenido de https://www.piranirisk.com/es/blog/gestion-de-riesgos-proyectos-de-software

Wikipedia, O. (s.f.). *Desarrollador de software*. Obtenido de https://es.wikipedia.org/wiki/Desarrollador_de_software