El futuro digital es de todos

MinTIC

SEGURIDAD DIGITAL WEB

Anexo 3

Resolución MinTIC 1519 del 2020
Condiciones mínimas técnicas y de seguridad digital

MinTIC – Viceministerio de Transformación Digital
Dirección de Gobierno Digital
Diciembre 2020

Karen Abudinen Abuchaibe - Ministra de Tecnologías de la Información y las Comunicaciones
German Rueda - Viceministro de Transformación Digital
Aura María Cifuentes - Directora de Gobierno Digital

Juan Pablo Salazar - Coordinación de Política
Juan Carlos Noriega - Coordinación de Política

Angela Janeth Cortés – Coordinación de Seguridad Digital
Senen Niño Gil – Coordinación de Seguridad Digital

Versión	Observaciones
Versión 1 Agosto/ 2020	Título Anexo 3 – Condiciones técnicas y de seguridad digital

Comentarios, sugerencias o correcciones pueden ser enviadas al correo electrónico: gobiernodigital@mintic.gov.co

Anexo 3 – Condiciones técnicas y de seguridad digital

Contenido

El presente anexo contiene las condiciones mínimas técnicas y de seguridad digital aplicables a los sujetos obligados en sus sitios web:

3.1 DEFINICIONES GENERALES.

Definiciones generales aplicables al presente anexo:

1. Cookies: Archivo creado por los sitios web en donde se almacenan datos sobre la navegación del usuario.

2. colCERT: Grupo de Respuesta a Emergencias Cibernéticas del Ministerio de Defensa Nacional.

3. CSIRT – Gobierno: Grupo de Respuestas a Incidentes de Seguridad Informática del Gobierno Nacional.

4. Defacement: Denominación en inglés que hace referencia al tipo de ataque cibernético, bajo el cual, se desconfiguran los contenidos del sitio o portal web, incluso poniéndoles "otra cara" mediante colores, imágenes o contenidos no originales.

5. Escape de variables en el código: Proceso de validación para confirmar que los datos ingresados en una variable correspondan con las entradas o caracteres válidos asignados por defecto en su configuración.

6. Hardening: En el contexto de seguridad digital, *hardening* o endurecimiento, implica eliminar todas las configuraciones por defecto, reduciendo las vulnerabilidades y asegurando los sistemas e infraestructuras digitales.

7. Plugins: Herramientas o aplicaciones de software que realizan funciones específicas, añadiendo en un software principal.

8. Políticas de origen de las cabeceras: Serie de reglas que garantizan la seguridad de las cabeceras del protocolo HTTP.

9. Script. Es un conjunto de comandos que se usa para la configuración del código fuente.

10. Token de CSRF: Código único de seguridad enviado en forma remota para validar la identidad del usuario.

3.2 CONDICIONES DE SEGURIDAD DIGITAL

Los sujetos obligados tendrán que adoptar medidas para garantizar la seguridad digital y mitigar riesgos de incidentes cibernéticos o filtración de datos personales o sensibles, observando lo siguiente:

1. Adoptar autónomamente políticas para implementar un sistema de gestión de seguridad digital y de seguridad de la información, conforme con las buenas prácticas internacionales. Entre otros podrán implementar los estándares de la familia ISO 27000 y/o los recomendados por el Instituto Nacional de Tecnología y Estándares (NIST, por sus siglas en inglés). Para cumplimiento de lo anterior se requiere la adopción del Modelo de Seguridad y Privacidad de la Información (MSPI) recomendado por la Dirección de Gobierno Digital del Ministerio de Tecnologías de la Información y las Comunicaciones.

2. Las entidades públicas del orden nacional y territorial, en caso de incidentes cibernéticos graves o muy graves, conforme con los criterios de su sistema de gestión de seguridad digital y seguridad de la información, deberán reportarlos por tardar dentro de las 24 horas siguientes a su detención al CSIRT-Gobierno. Para el resto de los sujetos obligados, deberán reportar al ColCERT del Ministerio de Defensa Nacional.

Adicional a lo anterior y de manera específica, los sujetos obligados deberán implementar los siguientes controles en el desarrollo de sitios web y aplicaciones:

1. Implementar controles de seguridad durante todo el ciclo de vida del desarrollo de software
2. Implementar o exigir controles de seguridad relacionados con el control de la autenticación, definición de roles y privilegios y separación de funciones
3. Exigir medidas de seguridad al proveedor del hosting (políticas de seguridad robustas y un nivel de madurez en seguridad optimizado).
4. Aplicar mecanismos de *hardening* para eliminar configuraciones y credenciales por defecto, además de deshabilitar métodos *HTTP* peligrosos como *put, delete, trace* y restringir en lo posible la administración remota.
5. Proteger la integridad del código, mediante: (i) la validación exhaustiva de: *inputs*, variables *post y get* (no enviar parámetros sensibles a través del método *get*), Cookies (habilitar atributos de seguridad como *Secure y HttpOnly*), y, cabeceras *HTTP*; (ii) la sanitización de los parámetros de entrada: es decir, que cuando se reciba la información de dichas variables

se eliminen etiquetas, saltos de línea, espacios en blanco y otros caracteres especiales que comúnmente conforman un *script*, además de la restricción de formatos y tamaños de subidas de archivos; (iii) la sanitización y escape de variables en el código; (iv) verificación estándar de las *Políticas de Origen de las cabeceras*; y (v) la verificación y comprobación del *token de CSRF* (cuando aplique).

6. Ejecutar monitoreos de seguridad sobre las páginas web que contemple, entre otras, las siguientes acciones: escaneo de archivos infectados, escaneo de vulnerabilidades, análisis de patrones para detectar acciones sospechosas, verificación contra listas negras, monitoreo del tráfico para detectar ataques de denegación de servicios.
7. Exigir mecanismos de autenticación dentro de los sitios web a través de la creación de contraseñas fuertes y solicitar renovaciones periódicas de las mismas garantizando la accesibilidad de persona con discapacidad.
8. Mantener actualizado el software, *frameworks y plugins* de los sitios web.
9. Restringir el uso de *login* contra ataques de fuerza bruta, implementando, entre otros: mecanismos de *captcha* accesibles o auto detectable, y/o limitar la tasa de intentos de *login*.
10. Ocultar y restringir páginas de acceso administrativo.
11. Restringir la escritura de archivos desde la web a través de la asignación de permisos de solo lectura.
12. Crear copias de respaldo.
13. Almacenar trazas o logs de auditoría de los eventos de seguridad, logins, entre otros.
14. Garantizar conexiones seguras a través de uso de certificados, *SSL (HTTPS* para la confianza de usuarios) y cifrado en la estructura de las peticiones para portales transaccionales, para evitar la manipulación de parámetros en las peticiones. (adicional al cifrado *SSL*), También deben habilitar las cabeceras de seguridad, entre otras las siguientes: *Content-Security-Policy (CSP), X-Content-Type-Options, X-Frame-Options, X-XSS-Protection, Strict-Transport-Security (HSTS), Public-Key-Pins (HPKP) Referrer-Policy, Feature-Policy.*
15. Implementar mensajes genéricos de error, que no revelen información acerca de la tecnología usada, excepciones o parámetros que dispararon el error específico, los cuales deberán ser comprensibles por parte de las personas, incluyendo la accesibilidad para las personas con discapacidad.
16. Proteger el binario de la aplicación, a través de métodos de ofuscación que impidan realizar procedimientos de ingeniería inversa (*reversing*) para analizar la lógica de la aplicación.
17. Sanitización de parámetros de entrada mediante la eliminación de etiquetas, saltos de línea, espacios en blanco y otros caracteres especiales que comúnmente conforman un «script», además de la restricción de formatos y tamaños para subida de archivos.
18. Sanitización de caracteres especiales (secuencia de Escape de variables en el código de Programación)

19. Revisar las recomendaciones de seguridad en la guía de desarrollo seguro de aplicaciones y Servicios Web Seguros de la *Open Web Application Security Project* (OWASP).
20. Implementar en los servidores los controles necesarios (hardware o software) de protección de acceso y de ataques como Cross-site scripting, SQL injection o Denial-of-service, entre otros.
21. Incorporar validación de formularios tanto del lado del cliente como del lado del servidor.
22. Implementar monitoreos de seguridad sobre la plataforma tecnológica que hace parte del sitio web (escaneo de vulnerabilidades, escaneo de archivos infectados, análisis de patrones para detectar acciones sospechosas, verificación contra listas negras, monitoreo del tráfico para detectar ataques de denegación de servicios) y realizar las acciones de mitigación correspondientes.
23. Establecer los planes de contingencia, DRP y BCP, que permita garantizar la continuidad de la sede electrónica o del sitio web 7/24 los 365 días del año.
24. Restringir la escritura de archivos en el servidor web a través de la asignación de permisos de roles y los privilegios asociados.
25. Implementar sistemas antivirus en el servidor web, para garantizar medidas contra infecciones de malware a los archivos del mismo.
26. Controlar el escalamiento de privilegios en los Sistemas Operativos, servidor web y Bases de datos que hacen parte de la infraestructura del portal web.

3.3 PROGRAMACIÓN DEL CÓDIGO FUENTE.

Los sujetos obligados, en todos sus sitios web, móvil y aplicaciones deberán implementar estándares de desarrollo seguro para evitar vulnerabilidades el código fuente y errores de presentación o alteraciones en el contenido de la información dispuesta al público. Así mismo, se deben evitar mecanismos que puedan poner en riesgo la información o los datos personales o sensibles.
Los sujetos obligados deben adoptar las siguientes buenas prácticas:

1. Realizar análisis estático del código con el objetivo de identificar vulnerabilidades que se encuentra en la programación de las aplicaciones.
2. Cumplir con la estandarización de código fuente para portales web, siguiendo las buenas prácticas del *W3C (World Web Wide Consortium*), de forma que permita la correcta visualización de la información a los usuarios.
3. Adoptar validadores *HTML y CCS* para la continua revisión del sitio web y su mejora continua, a través de las buenas prácticas del *W3C (World Web Wide Consortium)*.

4. Cumplir con los estándares definidos para la integración al Portal Único del Estado Colombiano GOV.CO, incluyendo la validación de la codificación, en caso de que les aplique.
5. Incluir lenguaje común de intercambio para la generación y divulgación de la información y datos estructurados y no estructurados dispuestos en medios electrónicos, como los sitios web de los sujetos obligados y el Portal Único del Estado Colombiano GOV.CO, en caso de que les aplique.
6. Implementar un sistema de control de versiones (Git), que permitan planear y controlar la vida de la aplicación, y en una fase a mediano plazo poder implementar un sistema de integración, cambio y despliegue continuo.