ALCALDÍA MAYOR DE BOGOTÁ - SECRETARÍA GENERAL

Rad. No. 2-2018-9331
Fecha: 25/04/2018 15:44:27
Destino: ENTIDADES DISTRITALES

Copia: N/A
Anexos: 3 FOLIOS

ALCALDÍA MAYOR DE BOGOTÁ D.C.
SECRETARÍA GENERAL

CIRCULAR N°. 026

Para: **SECRETARIOS(AS) DE DESPACHO, DIRECTORES(AS) DE DEPARTAMENTOS ADMINISTRATIVOS, GERENTES Y DIRECTORES(AS) DE UNIDADES ADMINISTRATIVAS ESPECIALES, GERENTES PRESIDENTES(AS) Y DIRECTORES(AS) DE ESTABLECIMIENTOS PÚBLICOS, EMPRESAS INDUSTRIALES Y COMERCIALES, SOCIEDADES PÚBLICAS, SOCIEDADES MIXTAS, ENTES UNIVERSITARIOS, HOSPITALES, ALCALDES LOCALES, VEEDOR DISTRITAL, CONTRALOR DISTRITAL Y LOCALES, DE BOGOTÁ, PERSONERO DE BOGOTÁ.**

De: **ALTA CONSEJERÍA DISTRITAL DE TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIONES.**

Asunto: **ASEGURAMIENTO EN LA COMPRA Y DESARROLLO DE APLICATIVOS, PÁGINAS WEB Y SOFTWARE EN GENERAL**

La Alta Consejería Distrital de Tecnologías de Información y Comunicaciones, en cumplimiento de su función de dirigir y liderar la formulación, articulación y seguimiento de las políticas, lineamientos y directrices distritales en materia de Tecnologías de Información y Comunicaciones para el fortalecimiento de la función administrativa y misional de los sectores y entidades de Bogotá Distrito Capital, se permite recalcar cómo impacta la seguridad y privacidad de la información cuando en la programación de las aplicaciones, páginas web y software en general, se dejan de lado los controles que permitan asegurar la misma, ya sea por parte de los fabricantes de software o proyectos misionales emprendidos al interior de la entidad.

Las aplicaciones, software y páginas web, pueden ser complejas cuando interactúan con múltiples sistemas y en las entidades la tarea de producir una aplicación segura o corregir una ya existente puede no ser fácil. Sin embargo la seguridad en las aplicaciones, software y páginas web no es opcional, en parte por el incremento de ataques y porqué debe darse el cumplimiento normativo.

Por lo anterior, se recomienda tener en cuenta en el desarrollo de software, aplicativos y web o en la adquisición del mismo, la implementación de buenas prácticas que impacten de forma positiva la seguridad y protección de la información al interior de la entidad.

Es importante resaltar, que se deben tener presentes y aplicar las guías G8_Controles_Seguridad y lo relativo a G15_Auditoria,propuestas por MinTIC para la implementación del Modelo de Seguridad y Privacidad de la Información -MSPI, disponibles en el enlace www.mintic.gov.co/gestionti/615/w3-propertyvalue-7275.html

De igual manera se recomienda consultar las mejores prácticas del dominio *Sistemas de Información* del marco de referencia de arquitectura empresarial, que se encuentra disponible en el enlace:www.mintic.gov.co/arquitecturati/630/articles-9266_recurso_pdf.pdf

En lo referente a la ley de protección de datos personales se debe tener presente lo promulgado en la Ley 1581 de 2012,y la Circular 2 expedida por la Alta Consejería Distrital de TIC en febrero de 2018. Siempre que se desarrolle se debe realizar un análisis de impacto en la intimidad sobre la forma cómo funcionará el nuevo aplicativo, software o pagina web.

En el siguiente Anexo, se hace un acercamiento a una buena práctica propuesta en el marco de referencia de arquitectura TI propuesto por MinTIC, que puede servir como ejemplo al momento de emprender desarrollos de aplicaciones, páginas web o software en general o en el caso de tomar la decisión de adquirirlo con proveedores especializados en el tema, para solicitar a estos los contemplen dentro de sus prácticas de desarrollo.

Las dudas respecto a dichos temas serán con gusto atendidas por la funcionaria María del Pilar Niño Campos a través del correo electrónico: mpninio@alcaldiabogota.gov.co o al teléfono 3813000 Ext. 3061.

Cordialmente,

SERGIO MARTÍNEZ MEDINA
Alto Consejero Distrital de TIC

Anexos: (3) folios.
Proyectó: María Del Pilar Niño Campos
Revisó: Iván Mauricio Hernández Lanao

ANEXO

CONTIENE METODOLOGÍA DESARROLLO SEGURO

INTRODUCCIÓN

Las aplicaciones, software y páginas web, pueden ser complejas cuando interactúan con múltiples sistemas y en las entidades la tarea de producir una aplicación segura o corregir una ya existente puede no ser fácil. Sin embargo la seguridad en las aplicaciones, software y páginas web no es opcional, en parte por el incremento de ataques y porqué debe darse el cumplimiento normativo.

En este entendido, la seguridad en el desarrollo de software, ha evolucionado y existen frameworks de trabajo y de buenas prácticas que orientan el deber ser.

Con el objetivo de definir estándares de desarrollo de código seguro, varias compañías de todo el mundo como Siemens en Alemania, Tata en India, Nomura Research en Japón, CIBC en Londres y las americanas Kaiser Permanente, Boeing, Cisco, Symantec, Intel y American Express, se unieron a esta iniciativa.

A este grupo de empresas se unieron también desarrolladores de otras organizaciones y universidades, como Amazon, Secure Compass, Universidad Carnegie Mellon y el equipo OWASP, para dar lugar al grupo de programación segura que se reúne para definir estándares y documentos que indiquen las habilidades necesarias para escribir código seguro.[1]

El grupo pretende mejorar los procesos en el desarrollo de software y tienen un patrón común "Los desarrollos dependen de la calidad del software", dentro de este contexto se han derivado diferentes buenas prácticas a fin de desarrollar software seguro .

En el siguiente Anexo, se hace un acercamiento a una de estas buenas prácticas en el marco de referencia de arquitectura TI propuesto por MinTIC, que puede servir como ejemplo al momento de emprender desarrollos de aplicaciones, páginas web o software en general o en el caso de tomar la decisión de adquirirlo con proveedores especializados en el tema, para solicitar a estos los contemplen dentro de sus prácticas de desarrollo.

[1] http://web.usbmed.edu.co/usbmed/fing/v3n1/v3n1a2.pdf

DESARROLLO DE MODELO SAMM

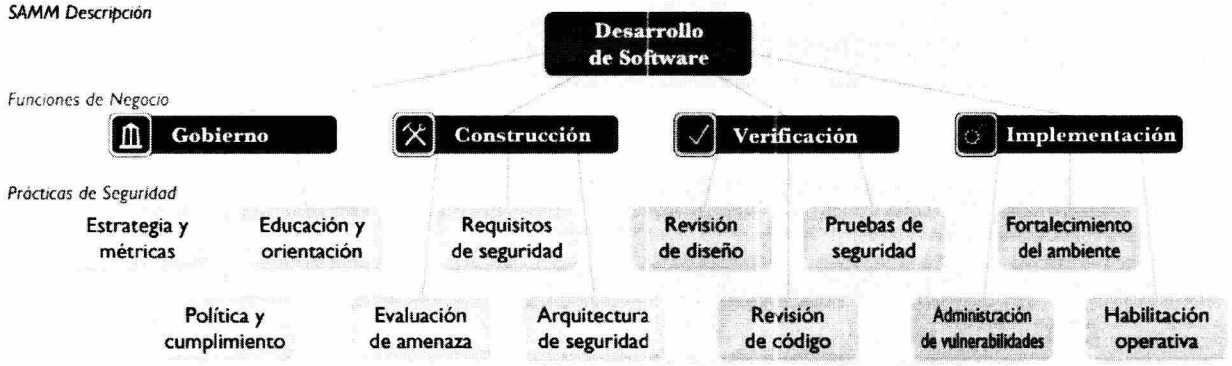
Lograr aplicaciones, páginas web, y software en general, seguras solo es posible cuando se utiliza un ciclo de desarrollo de software seguro, para lo cual OWASP (Proyecto abierto de seguridad en aplicaciones Web)[2] recomienda que las organizaciones establezcan una base sólida de formación, estándares y herramientas que hagan posible la codificación segura.

OWASP es una comunidad libre y abierta enfocada en mejorar la seguridad de los programas y aplicativos, y dentro de las estrategias proponen el uso del modelo SAMM (*Software Assurance Maturity Model*), el cual es un marco de trabajo abierto que ayuda a definir y estructurar una estrategia de seguridad en el desarrollo de *software* basada en los riesgos específicos que enfrenta cada organización, de acuerdo a las funciones críticas del negocio.

El modelo SAMM[3] fue diseñado para ser flexible, de tal manera que puede ser adoptado por cualquier tamaño de organización.

SAMM precisa 3 prácticas de seguridad por cada una de las funciones de negocio:

Figura 1. Modelo SAMM



Fuente: http://www.opensamm.org/download/

I. Prácticas de seguridad de la función de gobernabilidad

- Estrategia y métricas de seguridad: Consiste en la definición de una estrategia del programa de aseguramiento de *software* y la definición de los procesos y actividades para la recolección de métricas de seguridad.

-

[2]https://www.owasp.org/index.php/Sobre_OWASP
[3]http://www.opensamm.org/downloads/SAMM-1.0-es_MX.pdf

- Políticas y cumplimiento: Involucra el establecimiento de lo que está permitido hacer por la aplicación y los usuarios de la aplicación. Identificar y trabajar dentro del ámbito de las políticas de seguridad mientras se diseña la aplicación, asegura un correcto despliegue y el cumplimiento de requerimientos regulatorios.

- Educación y orientación: Se refiere a incrementar el conocimiento de seguridad del personal involucrado en el SDLC a través de un plan de capacitación y concientización de acuerdo a las funciones de los diferentes actores. Algunos de los tópicos a incluir son: manejo de errores, administración de bitácoras, autenticación, autorización, entre otros.

II. Prácticas de seguridad de la función de construcción

- Evaluación de amenazas: Esta práctica se centra en la identificación y entendimiento de los riesgos de la aplicación. De acuerdo al OWASP los 10 riesgos más importantes de seguridad en desarrollo de software, aplicaciones y páginas web son:

  - Inyección de código.
  - *Cross site scripting* (XSS).
  - Pérdida de autenticación y gestión de sesiones.
  - Referencia directa insegura a objetos.
  - Falsificación de peticiones en sitios cruzados (CSRF).
  - Configuración de seguridad defectuosa.
  - Almacenamiento criptográfico inseguro.
  - Falla de restricción de acceso a URL.
  - Protección insuficiente en la capa de transporte.
  - Redirecciones y reenvíos no validados.

- Requerimientos de seguridad: Se refiere a las expectativas de la aplicación respecto a la seguridad; los requerimientos de seguridad deben estar basados en las necesidades del negocio. Algunos de los requerimientos de seguridad que se deben considerar en el desarrollo de aplicaciones, software y páginas Web son:

  - Autenticación de usuarios.
  - Autorización de usuarios.
  - Prevención de manipulación de parámetros.
  - Protección de datos sensibles.
  - Prevención de *hacking* de sesión.
  - Validación de datos de entrada.
  - Auditoría y registro de actividades y transacciones.
  - Cifrado y *hashing* de datos confidenciales.

- Arquitectura de seguridad: Se refiere a introducir la seguridad en las aplicaciones web desde el diseño. La privacidad por diseño busca ir más allá del cumplimiento de las normas de protección de datos. Estas normas son un desarrollo del modelo de los principios prácticos de información justa o fair information practices principles (FIPP). La diferencia entre el modelo FIPP y el de privacidad por diseño es que, en el primero, la intimidad era una carga, mientras que en la privacidad por diseño resulta ser un buen "negocio", una característica que las personas quieren ver en los sistemas, procesos o servicios que las TIC facilitan. [4]

III. Prácticas de seguridad de la función de verificación

- Revisión de diseño: Se enfoca a la evaluación del diseño del *software* y problemas relacionados a la arquitectura, lo cual permite detectar problemas de manera temprana en el proceso de desarrollo de la aplicación, antes de que sea liberada.

- Revisión de código: Se refiere al proceso de revisar manualmente el código fuente de la aplicación para detectar huecos de seguridad. Existen numerosos problemas de seguridad en el desarrollo de aplicaciones, software y páginas Web, como los errores de inyección, que son mucho más fáciles de encontrar a través de revisión de código, que mediante pruebas externas. La revisión de código se debe realizar contra una lista de verificación que incluya:

  - Requerimientos del negocio acerca de la disponibilidad, integridad y confidencialidad.
  - Revisión del "Top 10 de OWASP".
  - Requerimientos específicos de la industria tales como Sarbanes-Oxley, ISO 17799, HIPAA, PCI, entre otros.
  - Algunas herramientas para la revisión de código como CodeCrawler, Orion y O2.

- Pruebas de seguridad: Involucra pruebas de seguridad para descubrir vulnerabilidades y establecer un estándar mínimo para la liberación del *software*. Así como la revisión de código tiene sus puntos fuertes, también los tienen las pruebas de seguridad. Es muy convincente cuando se puede demostrar que una aplicación es insegura demostrando su explotabilidad.

[4] https://karisma.org.co/que-es-la-privacidad-por-diseno-y-por-que-deberia-importarle/

IV. Prácticas de seguridad de la función de implementación

- Administración de vulnerabilidades: Involucra el establecimiento de procesos para manejo de vulnerabilidades e incidentes de seguridad, así como la recolección de métricas e información detallada que permita un análisis de la causa raíz del incidente para tomar acciones de mitigación.

- Fortalecimiento de los ambientes: La práctica se centra en el aseguramiento de la infraestructura que soporta a la aplicación Web, tales como: sistemas operativos, *firewalls* y manejadores de bases de datos.

- Habilitación operativa: La práctica se centra en la recopilación de información crítica por parte del equipo de desarrollo de la aplicación y la distribución de la misma a los usuarios y operadores. Abarca la documentación operativa y funcional de la aplicación.

Igualmente es vital tener presente los siguiente directrices de programación segura de forma general, a fin de mitigar los riesgos provenientes de un desarrollo inadecuado teniendo en cuenta que nunca se debe poner en riesgo la seguridad frente a la funcionalidad:

1. *Validaciones de entrada*
2. *Codificación de salidas*
3. *Manejo de errores y logs*
4. *Practicas criptográficas*
5. *Manejo de memoria*
6. *Estandarización y reutilización de funciones.*