M-CIB-009
Metodología General Para El Desarrollo De Aplicaciones Seguras

Responsables

Elaboró:	Especialista Ciberinteligencia
Revisó:	Control de Documentos
Aprobó:	Dirección General

Control de versiones

Versión	Fecha	Descripción del cambio
1	29/09/2022	Emisión Inicial

Clave del formato de manual: F-SGI-004 v3
Comentarios o dudas: sgi@silent4business.com

Contenido

1. Introducción 3
2. Alcance 3
3. Definiciones 3
4. Descripción del manual 4
A. Actividades realizadas por fase de la metodología 5
1. Definición de requerimientos de seguridad 5
2. Modelado de amenazas de la aplicación 5
3. Diseño de la arquitectura de seguridad 6
4. Ejecución de análisis de seguridad estático (SAST) 6
5. Ejecución de análisis de seguridad dinámico (DAST) 7
B. Herramientas por fases de la metodología 8
5. Anexos 12

1. Introducción

Silent4Business ha alineado los procesos y revisiones técnicas a metodologías mundialmente reconocidas como MS Security Development Lifecycle, NIST 800-160 y OWASP CLASP, así como OWASP Code Review Guide, SANS 542 y el OSSTMM. El equipo de Silent4Business utiliza las metodologías mencionadas para realizar el servicio de Acompañamiento de desarrollo seguro de aplicaciones.

A continuación, se muestra de manera general la metodología empleada para el Acompañamiento de desarrollo seguro de aplicaciones, la cual contempla la definición de requerimientos de seguridad del aplicativo, el modelado de amenazas de la aplicación, el diseño de la arquitectura de seguridad, y la realización de pruebas estáticas durante la fase de desarrollo del aplicativo, así como las pruebas dinámicas previo a desplegar la aplicación en ambientes de producción.

2. Alcance

Este proceso es de uso del área de ciberinteligencia, es aplicable para servicios ejecutados a clientes externos y/o para Silent4Business.

3. Definiciones

Activo

Recurso del sistema de información o relacionado con éste, necesario para que la organización funcione correctamente y alcance los objetivos propuestos.

Amenaza

Es un evento que puede desencadenar un incidente en la organización, produciendo daños materiales o pérdidas inmateriales en sus activos.

Ataque

Evento, exitoso o no, que atenta sobre el buen funcionamiento del sistema.

Comprometer

Evasión de controles de seguridad causando posible de pérdida o daño en un activo de cómputo.

Exploit

Código o técnica que es usada por una amenaza para tomar ventaja de una vulnerabilidad.

Impacto

Medición de la consecuencia al materializarse una amenaza.

Mitigación

Acciones para remediar vulnerabilidades identificadas en los activos.

Probabilidad

Es la posibilidad que existe entre una amenaza y las variables de entorno para que una vulnerabilidad sea aprovechada.

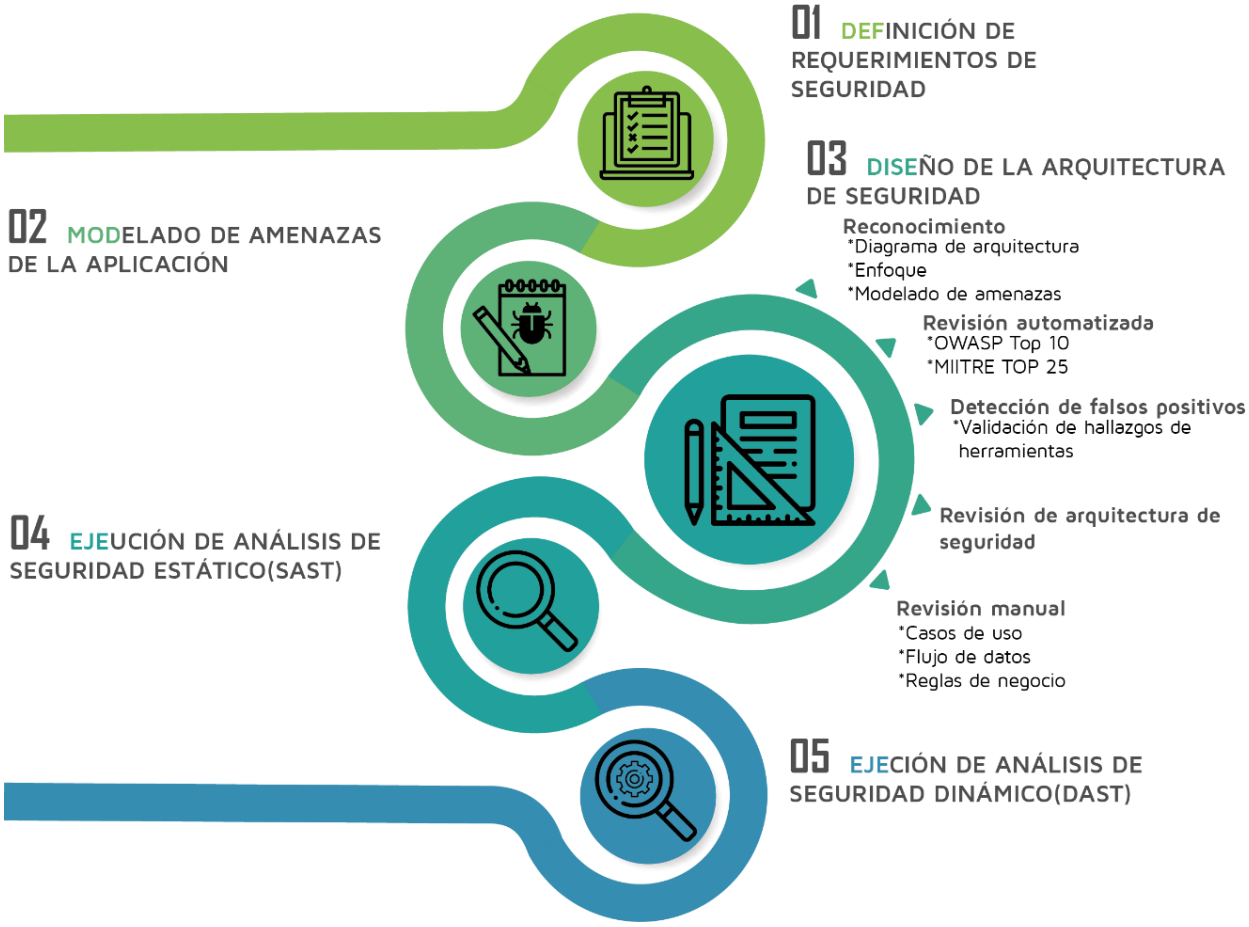
Riesgo

Es la probabilidad de que una amenaza explote una vulnerabilidad con su correspondiente impacto al negocio.

Vulnerabilidad

Son debilidades que influyen negativamente en un activo y que posibilita la materialización de una amenaza.

4. Descripción del manual



A. Actividades realizadas por fase de la metodología

Por cada fase de la metodología, el equipo de SILENT4BUSINESS está capacitado para realizar actividades específicas al desarrollo de aplicaciones seguras. A continuación, se detallan algunas de las actividades y/o técnicas empleadas:

1. Definición de requerimientos de seguridad

- Requerimientos de identificación
- Requerimientos de autenticación
- Requerimientos de autorización
- Requerimientos de inmunidad
- Requerimientos de integridad
- Requerimientos de detección de intrusos
- Requerimientos de protecciones físicas
- Requerimientos de auditoría de seguridad
- Requerimientos de no repudiación
- Requerimientos de privacidad

2. Modelado de amenazas de la aplicación

- Seccionar la aplicación
    - Dependencias externas
    - Puntos de entrada
    - Activos
    - Superficie de ataque
    - Niveles de confianza
    - Análisis de flujo de datos
    - Análisis de transacciones
    - Diagramas de flujo de datos
- Determinar y categorizar amenazas
    - STRIDE (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privileges)
    - ASF (Application Security Frame)
    - DREAD (Damage, Reproducibility, Exploitability, Affected Users, Discoverability)
- Determinar contramedidas y mitigaciones
    - Perfilar la amenaza
        - Amenazas no mitigadas
        - Amenazas parcialmente mitigadas
        - Amenazas completamente mitigadas
- Métricas

- Líneas de código
- Puntos funcionales
- Densidad de defectos
- Densidad de riesgos

3. Diseño de la arquitectura de seguridad

- Validación de campos de entrada
- Autenticación
- Autorización
- Gestión de configuración
- Gestión de sesión
- Criptografía empleada
- Manipulación de parámetros
- Manejo de excepciones
- Auditoria y registro de eventos
- Librerías y frameworks

4. Ejecución de análisis de seguridad estático (SAST)

- Reconocimiento
  - Revisión de diagrama de arquitectura
  - Definición de enfoque
  - Modelado de amenazas
- Revisión automatizada
  - Inyecciones
  - Vulnerabilidades de autenticación y de gestión de sesión
  - Cross-Site Scripting
  - Referencias inseguras a objetos
  - Malas configuraciones de seguridad
  - Exposición de datos sensibles
  - Fallas de Control de accesos
  - Cross-Site Request Forgery (CSRF)
  - Uso de componentes con vulnerabilidades conocidas
  - Redireccionamientos no validados
- Detección de falsos positivos
- Revisión manual
  - Validación de reglas de negocio
  - Revisión de flujo de datos
  - Validación de casos de uso
- Revisión de arquitectura de seguridad

5. Ejecución de análisis de seguridad dinámico (DAST)

- Reconocimiento
  - Identificación de tecnología en servidor web
  - Identificación de tecnología en aplicación
  - Cabeceras HTTP configuradas
  - Suites de cifrado empleadas
- Descubrimiento
  - Identificación de puntos de entrada
  - Mapeo de flujos de ejecución
  - Métodos habilitados
  - Política de contraseñas
  - Cookies de sesión del aplicativo
- Enumeración
  - Directorios del aplicativo
  - Roles de usuario del aplicativo
  - Cuentas válidas
  - Credenciales débiles
  - Manejo de errores
- Análisis de vulnerabilidades
  - Directorio transversal
  - Inclusión de archivos
  - Escalación de privilegios
  - Referencias inseguras a objetos
  - Cross-Site Scripting
  - Inyecciones SQL
  - Inyección de comandos
  - Inyecciones XML
  - Carga de archivos maliciosos
  - Vulnerabilidades de lógica de negocio

B. Herramientas por fases de la metodología

A continuación, se listan las herramientas que usa el equipo de SILENT4BUSINESS durante las diferentes fases de la metodología.

Fase de la Metodología	Herramienta	Descripción
Definición de requerimientos de seguridad	**BDD-security**	BDD-Security es un marco de prueba de seguridad que utiliza lenguaje natural en una sintaxis Gherkin Given, When, Then para describir los requisitos de seguridad como características. Esos mismos requisitos también son ejecutables como pruebas estándar de unidad / integración, lo que significa que pueden ejecutarse como parte del proceso de compilación / prueba / implementación.
Modelado de amenazas de la aplicación	**Microsoft Threat Modeling Tool**	Threat Modeling Tool es un elemento básico del Ciclo de vida de desarrollo de seguridad (SDL) de Microsoft. Permite a los arquitectos de software identificar y mitigar los posibles problemas de seguridad en una fase temprana, cuando son relativamente sencillos y poco costosos de resolver. En consecuencia, reduce en gran medida el costo total de desarrollo.
	ThreatModeler	ThreatModeler está basado en la metodología VAST para modelado de amenazas. VAST se refiere a una vista más moderna de los estándares de seguridad de las empresas: • Visual • Ágil • Simple • Modelado de amenazas
	VisualParadigm	La herramienta de modelado de amenazas de VP Online es una herramienta de modelado de amenazas basada en la web, con una interfaz de arrastrar y soltar para crear modelos de amenazas sin esfuerzo. Viene con todos los elementos estándar necesarios para crear un modelo de amenaza para varias plataformas.
	IriusRisk	IriusRisk es una herramienta que genera modelos de amenazas y su lista de requisitos de seguridad. Brinda un enfoque de autoservicio para administrar los requisitos de seguridad del software, al mismo tiempo que aplica las contramedidas estandarizadas y las políticas de seguridad acordadas por el equipo de seguridad.
Diseño de la arquitectura de seguridad	**Visio**	Microsoft Visio es un software de dibujo vectorial para Microsoft Windows. Las herramientas que lo componen permiten realizar diagramas de oficinas, diagramas de bases de

Fase de la Metodología	Herramienta	Descripción
		datos, diagramas de flujo de programas, UML, y más.
Ejecución de análisis de seguridad estático	**Checkmarx**	Checkmarx SAST (CxSAST) es una solución de análisis estático flexible y precisa de nivel empresarial que se utiliza para identificar cientos de vulnerabilidades de seguridad en código personalizado. Es utilizado por los equipos de desarrollo, DevOps y seguridad para escanear el código fuente temprano en el SDLC, identificar vulnerabilidades y proporcionar información procesable para remediarlos. Admite más de 22 lenguajes de codificación y secuencias de comandos y sus marcos con configuración cero para escanear cualquier idioma.
	SonnarQube	SonarQube® es una herramienta de revisión automática de código para detectar errores, vulnerabilidades y olores de código en su código. Se puede integrar con su flujo de trabajo existente para permitir la inspección continua de código en las ramas de su proyecto y las solicitudes de extracción.
	Veracode	El análisis estático de Veracode le permite identificar y remediar rápidamente las fallas de seguridad de la aplicación a escala y eficiencia. Nuestra plataforma basada en SaaS se integra con sus herramientas de desarrollo y seguridad, haciendo que las pruebas de seguridad sean una parte perfecta de su proceso de desarrollo. Una vez que se identifiquen las fallas, aproveche los consejos de remediación en línea y el asesoramiento individual para reducir su resolución de tiempo medio. El análisis estático de Veracode es la ventaja competitiva que necesita para llevar sus aplicaciones al mercado de forma segura a la velocidad de DevOps.
	FindBugs	FindBugs es un programa para encontrar errores en programas Java. Busca instancias de "patrones de errores" --- instancias de código que probablemente sean errores.
	Micro Focus Fortify	Cubre todos los aspectos tales como pruebas de seguridad de las aplicaciones, la gestión de la seguridad del software y la protección automática de las aplicaciones para asegurar el software que apalanca al negocio. Fortify ofrece soluciones de seguridad de aplicaciones on-premis y on-demand para cubrir todas sus necesidades de seguridad de software, incluyendo la seguridad de Software, aplicaciones móviles y web.
	HCL Appscan Source	AppScan Source ayuda a las organizaciones a desarrollar software más seguro y a evitar vulnerabilidades costosas que surgen al final del ciclo de vida del desarrollo. Al integrar las pruebas de seguridad al inicio del ciclo de desarrollo, es decir, la seguridad de desplazamiento a la izquierda, AppScan reduce la exposición al riesgo y los costos de remediación. AppScan

Fase de la Metodología	Herramienta	Descripción
		Source utiliza su tecnología Intelligent Finding Analytics (IFA) basada en el aprendizaje automático para ayudar a los clientes a identificar rápidamente las vulnerabilidades de seguridad críticas y las mejores medidas para la corrección. Como resultado, se evitan remediaciones costosas al final del ciclo de desarrollo o en la producción.
	Técnicas manuales	Se emplean técnicas manuales para realizar la identificación de vulnerabilidades en el código; estas actividades están principalmente relacionadas con la búsqueda de vulnerabilidades que afectan la lógica de negocio; así como vulnerabilidades en código complejo.
Ejecución de análisis de seguridad dinámico	**Burp Suite**	Burp Suite es el software de prueba de seguridad de aplicaciones web más utilizado en el mundo. Burp viene en dos versiones: Burp Suite Professional para testers prácticos y Burp Suite Enterprise Edition con automatización escalable e integración de CI.
	SQLMap	SQLmap es una herramienta de prueba de penetración de código abierto que automatiza el proceso de detección y explotación de fallas de inyección SQL y toma de control de los servidores de bases de datos. Viene con un potente motor de detección, muchas funciones de nicho para el último probador de penetración y una amplia gama de interruptores que duran desde la toma de huellas digitales de la base de datos, la obtención de datos de la base de datos, hasta el acceso al sistema de archivos subyacente y la ejecución de comandos en el sistema operativo a través de conexiones fuera de banda.
	Nikto	Nikto es un escáner de servidor web de código abierto (GPL) que realiza pruebas exhaustivas contra servidores web para múltiples elementos, incluidos más de 6700 archivos / programas potencialmente peligrosos, verifica versiones obsoletas de más de 1250 servidores y problemas específicos de versión en más de 270 servidores. También verifica los elementos de configuración del servidor, como la presencia de múltiples archivos de índice, las opciones del servidor HTTP, e intentará identificar los servidores web y el software instalados. Los elementos de escaneo y los complementos se actualizan con frecuencia y se pueden actualizar automáticamente.

Fase de la Metodología	Herramienta	Descripción
	OWASP Zap	Zed Attack Proxy (ZAP) es una herramienta gratuita de prueba de penetración de código abierto que se mantiene bajo el paraguas del Open Web Application Security Project (OWASP). ZAP está diseñado específicamente para probar aplicaciones web y es flexible y extensible. En esencia, ZAP es lo que se conoce como un "proxy del hombre en el medio". Se interpone entre el navegador del probador y la aplicación web para que pueda interceptar e inspeccionar los mensajes enviados entre el navegador y la aplicación web, y modificar el contenido. si es necesario, y luego reenviar esos paquetes al destino. Se puede usar como una aplicación independiente y como un proceso de demonio.
	Xenotix	OWASP Xenotix XSS Exploit Framework es un marco avanzado de detección y explotación de vulnerabilidades Cross Site Scripting (XSS). Se incorpora con un módulo de recopilación de información rico en funciones para el reconocimiento de objetivos. El Exploit Framework incluye módulos de explotación XSS altamente ofensivos para pruebas de penetración y creación de prueba de concepto.
	SSLScan	Es una escáner de puertos SSL la cual nos facilita información sobre qué tipo de cifrado soporta el puerto al que nos conectamos, que tipo de cifrado es el preferido, que protocolos SSL están soportados, información sobre el certificado instalado, permitiéndonos una salida a fichero en formato XML con el que, posteriormente, se pueden elaborar informes.
	Nmap	Nmap es una utilidad para el descubrimiento de redes y la auditoría de seguridad. Nmap utiliza paquetes IP sin procesar de formas novedosas para determinar qué hosts están disponibles en la red, qué servicios (nombre y versión de la aplicación) están ofreciendo, qué sistemas operativos (y versiones de SO) están ejecutando, qué tipo de paquetes de filtros / firewalls están en uso, y docenas de otras características. Fue diseñado para escanear rápidamente redes grandes, pero funciona bien con hosts únicos.
	Nessus	Nessus es un programa de escaneo de vulnerabilidades en diversos sistemas operativos. Consiste en un demonio o diablo, nessusd, que realiza el escaneo en el sistema objetivo, y nessus, el cliente (basado en consola o gráfico) que muestra el avance e informa sobre el estado de los escaneos. Desde consola nessus puede ser programado para hacer escaneos

Fase de la Metodología	Herramienta	Descripción
		programados con cron.
	DotDotPwn	Es un fuzzer inteligente muy flexible para descubrir vulnerabilidades de directorio transversal en software como servidores HTTP/FTP/TFTP, plataformas web como CMS, ERP, blogs, etc. Además, tiene un módulo independiente del protocolo para enviar el payload deseado al host y al puerto especificado. Por otro lado, también se podría usar de forma secuencial utilizando el módulo STDOUT. Está escrito en lenguaje de programación perl y se puede ejecutar en plataformas OS X, *NIX o Windows. Es la primera herramienta mexicana incluida en BackTrack Linux (BT4 R2).

5. Anexos

NA