

Artefactos de seguridad para el monitoreo de seguridad orientada a aplicaciones web

**Breve descripción:**

En esta última década el avance tecnológico impulsó a las empresas a trasladar buena parte de sus actividades y servicios hacia el mundo digital por medio de la sistematización de sus procesos a través de la creación de aplicaciones web. Sin embargo, pese a las ventajas adquiridas, también se contraen sus principales desventajas ligadas a los ataques cibernéticos, virus y demás afectaciones posibles del mundo “software”.

**Octubre 2023**

Tabla de contenido

[Introducción 1](#_Toc147760435)

[1. Seguridad en Aplicaciones Web 3](#_Toc147760436)

[Plan Estratégico de Seguridad en Aplicaciones Web 5](#_Toc147760437)

[2. Fases para la adopción de un Plan Estratégico de Seguridad en Aplicaciones Web 7](#_Toc147760438)

[2.1. Identificación del estado actual de la Seguridad en Aplicaciones Web 7](#_Toc147760439)

[Niveles de verificación de seguridad de aplicaciones web 9](#_Toc147760440)

[Ejemplos en la industria 11](#_Toc147760441)

[2.2. Definición de los objetivos de la Seguridad en Aplicaciones Web 12](#_Toc147760442)

[2.3. Determinación del estado deseado de la Seguridad en Aplicaciones Web 13](#_Toc147760443)

[2.4. Definición de los Indicadores en Aplicaciones Web 14](#_Toc147760444)

[Indicadores Claves de Rendimiento (KPI, “Key Performance Indicator”) 20](#_Toc147760445)

[2.5. Definición de Métricas de Seguridad en Aplicaciones Web 23](#_Toc147760446)

[3. Ejecución del Plan Estratégico de la Seguridad en Aplicaciones Web 26](#_Toc147760447)

[3.1. Herramientas de Monitoreo de Seguridad en Aplicaciones Web 28](#_Toc147760448)

[3.2. El marco de conocimiento de seguridad (“Security Knowledge Framework”) 29](#_Toc147760449)

[OWASP ZAP (“Zed Attack Proxy”) 30](#_Toc147760450)

[OWASP “Juice Shop” 31](#_Toc147760451)

[Demos herramientas “Juice Shop” y OWASP ZAP 32](#_Toc147760452)

[3.3. Medición de tráfico de red 33](#_Toc147760453)

[Wireshark 34](#_Toc147760454)

[NTOP 35](#_Toc147760455)

[Microsoft Message Analyzer 37](#_Toc147760456)

[“Paessler Router Traffic Grapher” (PRTG) 38](#_Toc147760457)

[Síntesis 41](#_Toc147760458)

[Material complementario 43](#_Toc147760459)

[Glosario 45](#_Toc147760460)

[Referencias bibliográficas 47](#_Toc147760461)

[Créditos 50](#_Toc147760462)

Introducción

Le damos la bienvenida al componente formativo denominado “Artefactos de seguridad para el monitoreo de seguridad orientada a aplicaciones web”, el cual hace parte del programa de formación Técnico en “Seguridad de aplicaciones web”, para lo cual se invita a observar el siguiente video:

1. Artefactos de seguridad para el monitoreo de seguridad orientada a aplicaciones web



[**Enlace de reproducción del video**](https://www.youtube.com/watch?v=B4b2zI6DSFk)

|  |
| --- |
| **Síntesis del video: Artefactos de seguridad para el monitoreo de seguridad orientada a aplicaciones web** |
| El contenido va a presentarnos un plan estratégico de seguridad de aplicaciones web, el cual consta de cinco fases. La primera fase busca identificar el estado actual de la seguridad en las aplicaciones web; el segundo paso consiste en definir los objetivos que tiene el plan estratégico; el tercer paso busca determinar el estado al que se quiere llegar con el proceso de seguridad en aplicaciones web; la cuarta fase de este plan consiste en definir los indicadores y las métricas, que es la parte más importante de esta competencia, también se van a revisar los requisitos de seguridad, dado que ellos permiten al aprendiz identificar qué métricas son adecuadas para todo el tema de seguridad; y la última fase del plan es la ejecución de un plan estratégico.  Luego se van a presentar algunas herramientas de monitoreo de la seguridad en las aplicaciones web. |

# Seguridad en Aplicaciones Web

Las últimas décadas presenciaron la masificación de internet y los dispositivos inteligentes, trayendo consigo el estallido de las aplicaciones sociales y la era de la conexión total, esta revolución, generó un sinnúmero de acontecimientos que modificaron la forma como las personas ven los dispositivos o herramientas tecnológicas; puesto que, ahora se le da mayor prioridad a las cuentas, usuarios y contactos por encima del “hardware”, dado que este, es fácil de cambiar o mejorar por un nuevo dispositivo.

Esto indica que para las personas el activo más importante es su información personal y esta premisa es un común denominador que se extiende a las organizaciones y empresas, puesto que, dicha información es sensible y vital para el buen desarrollo de las actividades diarias, de los procesos pertinentes y en la prestación de sus servicios. Esto claramente indica la innegable dependencia de la tecnología y del activo informático, siendo los soportes claros de los sistemas de una organización. No obstante, no ha sido siempre de esta manera.

Años atrás para las empresas era más sencillo proteger sus activos, ya que las grandes plataformas de datos se encontraban aisladas e independientes de conectividad a la red, usando arquitecturas centralizadas con terminales con capacidades de procesamiento muy limitadas. Empero, actualmente, la diversificación de las redes de telecomunicaciones y la masiva creación de plataformas externas hacen que la información organizacional quede cada vez más expuesta a todas las ciberamenazas y ataques informáticos (V. Mouli and K. P. Jevitha, 2016), en vista de que cualquier dispositivo conectado a internet es un foco sensible al ataque por las latentes vulnerabilidades, como un diseño inicial inadecuado, la ejecución de programas que pueden presentar rutinas de código realizadas con desidia y descuido, escasa implementación de medidas de control de acceso, falta de validación y verificación de datos de ingreso o inclusive el riesgo más latente originado por el error humano que es la ingeniería social (M. Seyyar, F. Çatak and E. Gül, 2018).

En (OWASP. Top Ten, 2022) es posible apreciar con mayor claridad las amenazas y vulnerabilidades más importantes a día de hoy para las organizaciones y sus aplicaciones web; a través del esfuerzo colaborativo global conocido como el Proyecto Abierto de Seguridad en Aplicaciones Web (OWASP, “Open Web Application Security Project”), que busca concientizar a los equipos de desarrollo y los gerentes de proyectos la importancia de proteger sus activos informáticos con la generación de procesos que estimen el nivel de seguridad de sus sistemas, de manera que sea posible tomar decisiones y medidas preventivas ante una ciberamenaza o una eventualidad.

Es así como:

En toda entidad las políticas e instrumentos de ciberseguridad son de carácter obligatorio y deben garantizar que se valoren los riesgos alineados con las tecnologías como amenazas reales a los procesos y servicios de la compañía, sin desestimar su impacto o sus consecuencias a corto, mediano y largo plazo para los objetivos trazados por la gerencia. Para ello, las organizaciones deben contar con procedimientos y personal especializado encargados de mitigar y reaccionar a las afectaciones que son producto de los riesgos en ciberseguridad.

El área de ciberseguridad, como eje transversal a la empresa debe crear o fortalecer los procesos de control y protección a todas las demás áreas de la organización, a fin de garantizar integridad, confidencialidad y la seguridad de su información y la de sus clientes. (A. Pineda, John A. Bohada y M. L. Pineda, 2018).

### Plan Estratégico de Seguridad en Aplicaciones Web

En (R. Rodríguez, H. Alberto, 2019) se describe el concepto de estrategia corporativa como: “el patrón de decisiones en una compañía, que determina y revela sus objetivos, propósitos o metas; que genera las principales políticas y planes para alcanzar dichas metas y define el rango de negocio que debe perseguir la compañía, el tipo de organización económica y humana que es o pretende ser y la naturaleza de la contribución tanto económica, como no económica, que pretende hacer a sus accionistas, empleados, clientes y comunidades”. Y en parte, esta idea puede ser transferida al contexto de un plan estratégico de seguridad en aplicaciones web.

El objetivo de un plan estratégico de seguridad en aplicaciones web, estriba en la definición específica de una ruta de trabajo que le permita a la organización llegar a la obtención de unas métricas reales capaces de establecer el rendimiento de manera cuantitativa; con el objetivo de lograr que las aplicaciones web lleguen a un estado de madurez y nivel deseado en materia de seguridad.

El trazado de una estrategia debe estar definido y alineado a un alto nivel con los objetivos de la organización y la operación del negocio. Por tanto, las pautas que se presentan a continuación son de común acuerdo entre la alta dirección y el equipo TI o los encargados del área de ciberseguridad; determinando el nivel de verificación de seguridad de aplicaciones al que desean llevar el sistema. Para ello, esta estrategia establece la base del plan de acción a ejecutar, permitiendo llevar el proceso de seguridad de las aplicaciones web desde el nivel actual hasta el nivel de madurez deseado. Dentro del plan de acción es apremiante conocer el personal y los recursos “hardware” y “software” disponibles, además de identificar las limitaciones existentes a nivel de conocimiento, posibilidad de adquisiciones, alcance y considerando un apartado para los requerimientos legales y regulatorios. Además, el plan estratégico debe establecer mecanismos de monitoreo y métricas bien definidas para determinar al final si el cambio en el nivel de verificación representa un éxito o un fracaso en el proceso.

El plan estratégico de seguridad en aplicaciones web debe establecer como mínimo los siguientes elementos abordados a continuación:

* **Un objetivo:** lo que se busca alcanzar a través de la ejecución de la estrategia.
* **Acciones:** actividades a ejecutar para cumplir con lo definido en la estrategia.
* **Resultados:** lo que se espera obtener de esas acciones que permiten cumplir con el objetivo propuesto.
* **Tiempos:** periodo en el cual se debe ejecutar la estrategia.
* **Indicadores:** una forma de medir y proporcionar una evaluación a los resultados.
* **Métricas:** el rendimiento real medido por los datos de manera cuantitativa.
* **Niveles:** el estado del sistema antes y después de la ejecución de la estrategia.

# Fases para la adopción de un Plan Estratégico de Seguridad en Aplicaciones Web

Las fases para la adopción de un Plan Estratégico de Seguridad en Aplicaciones Web son:

* Identificación del estado actual de la seguridad en aplicaciones web.
* Definición de los objetivos de la seguridad en aplicaciones web.
* Determinación del estado deseado de la seguridad en aplicaciones web.
* Definición de los indicadores y métricas de la seguridad en aplicaciones web.
* Ejecución del plan estratégico de la seguridad en aplicaciones web.

A continuación, se describe el objetivo de cada fase, los artefactos de entrada, de salida y actividades a ejecutar para cumplir cada uno de ellos.

## Identificación del estado actual de la Seguridad en Aplicaciones Web

El primer paso para generar un plan estratégico consiste en identificar el estado actual en el que se encuentra la seguridad en aplicaciones web; para ello es posible utilizar dos enfoques: a partir de un modelo de madurez o por los niveles ASVS que verifican la seguridad en sistemas web. El enfoque por madurez requiere de un proyecto base que permita generar el contraste y el punto de partida, por ello el enfoque basado en los niveles permite iniciar el plan a partir del esfuerzo en conjunto de la dirección general y los equipos relacionados con TI siguiendo el marco de referencia ofrecido por el OWASP y sus comunidades. Para ello se tienen los siguientes artefactos de entrada, salida y las actividades en esta fase:

1. **Artefactos de entrada**

* Inventario de activos “software”, licencias y “hardware”.
* Inventario de activos de información.
* Inventario de personal especializado o con potencial técnico.

Actividades

* Reunir todos los artefactos de entrada solicitados, documentación de las actuales estrategias de la organización e iniciativas sobre este tema.
* Recopilar las mejores prácticas externas de expertos en la materia.

1. **Artefactos de salida**

* Análisis del estado actual de los sistemas.
* Lista de chequeo de requisitos de seguridad específica para su aplicación basado en los niveles ASVS.

Actividades

* Realizar una validación de todo el inventario recogido y comparar con los niveles de verificación de seguridad ASVS.
* Identificar el nivel de verificación de seguridad ASVS.
* Generar los artefactos de salida a partir del análisis realizado en la actividad anterior.

### Niveles de verificación de seguridad de aplicaciones web

El ASVS ha definido tres niveles de verificación en los cuales se aumenta la profundidad de los conceptos con cada nivel superado y estos contienen una lista de requisitos intrínsecos que garantizan la ciberseguridad correspondiendo a una funcionalidad o un grupo de ellas orientadas netamente al tema en cuestión, las cuales deben ser implementadas por los equipos de desarrollo de “software”. Los niveles son los siguientes:

* **Nivel 1 (Oportunista).** El nivel base no distingue “software”, puesto que sirve para todos los sistemas.
* **Nivel 2 (Estándar).** Se encuentra conducido para aplicaciones que dentro de su operación manejan datos sensibles que requieren un nivel de protección.
* **Nivel 3 (Avanzado).** Este nivel es para sistemas críticos encargados de realizar transacciones de alto valor o dentro de su información existen datos personales sensibles y confidenciales, que son el más alto nivel de confianza.

A continuación se profundiza en qué consiste cada nivel:

* **Nivel 1 (Oportunista).** Para una organización, el nivel oportunista es muy apropiado si tiene sistemas donde la información capturada requiere de un nivel bajo de confiabilidad e integridad, donde la falta o pérdida de algunos de estos datos no impacte inminentemente el proceso o flujo de la aplicación.

Este nivel proporciona a la gerencia un análisis rápido al ecosistema de aplicaciones que posean o les permite elaborar una lista de requerimientos en seguridad de aplicaciones que hacen parte de un plan estratégico de seguridad realizado en fases.

* **Nivel 2 (Estándar).** El nivel dos del ASVS tiene como premisa asegurar que dentro de las organizaciones se conozcan y existan controles de seguridad en las aplicaciones ubicados en lugar correspondiente y con el mayor porcentaje de efectividad.

Las aplicaciones que deben estar a este nivel por lo general realizan transacciones comerciales entre organizaciones conocidas como negocio a negocio (“business-to-business”, B2B), sistemas de gestión de información del área de la salud, aplicaciones que implementan operaciones sensibles para el negocio o mantienen otros activos sensibles de terceros.

* **Nivel 3 (Avanzado).** Las aplicaciones a este nivel deben estar modularizadas considerablemente buscando facilitar la respuesta rápida ante fallos, la escalabilidad y una defensa coral construida de varias capas de seguridad. Cada módulo podría estar separado físicamente o en diferente conexión de red, a fin de lograr que por sí mismos se responsabilicen por la seguridad, estando todo debidamente documentado.

Dentro de las responsabilidades obligatoriamente se incluyen controles que certifican la confidencialidad a través del cifrado de los datos, la integridad transaccional, la validación de entradas, el manejo de auditorías en capa de persistencia, entre otros.

### Ejemplos en la industria

Como se apreció anteriormente las diferentes amenazas son motivadas por distintos factores y propósitos, puesto que existen algunos sectores en la industria con activos de información únicos y valiosos para los atacantes, los cuales podrían causar gran conmoción con su revelación o simplemente a partir de ellos es podrían obtener beneficios de cualquier índole, obligando explícitamente a estas organizaciones a cumplir con regulaciones y normas específicas.

No obstante, es claro que cada sector posee criterios únicos y bien diferenciados en cuanto a las amenazas y vulnerabilidades presentes, es común en todos los segmentos de la industria encontrar ataques oportunistas. En este tipo se busca cualquier sistema que se encuentre en un estado vulnerable para explotar sus debilidades de una manera rápida y sencilla. Por lo que el rango oportunista se recomienda para todas las aplicaciones y debe ser el punto base para manejar las amenazas más habituales de encontrar. Lo invitamos a ver el PDF **Ejemplos por sector de la industria**, el cual se encuentra en la carpeta Anexos, donde se identifican algunas recomendaciones para cada nivel en diferentes sectores de la industria.

## Definición de los objetivos de la Seguridad en Aplicaciones Web

La segunda fase en el plan estratégico consiste en definir y acordar el alcance a través de unos objetivos claros alineados a los objetivos de la organización. Puesto que, la compañía actúa como “stakeholder” buscando que el plan presentado satisfaga las necesidades establecidas por la gerencia y permitan mejorar sustancialmente sus procesos con un nivel de verificación superior en sus aplicaciones. En seguida se aprecian los artefactos de entrada, salida y las actividades en esta fase:

1. **Artefactos de entrada**

* Plan estratégico institucional.
* Lista de chequeo fase anterior.
* Limitaciones de negocio o de recursos de TI que afecten el logro de los objetivos.
* ASVS, OWASP y otras listas similares.

Actividades

* Realizar talleres entre las partes interesadas para establecer los objetivos del plan estratégico y la meta a la que se anhela llegar en el nivel de verificación en seguridad de las aplicaciones web.
* Generar listas de chequeo de requerimientos de seguridad sobre los niveles ASVS.

1. **Artefactos de salida**

* Objetivos del plan estratégico de seguridad en aplicaciones web.
* Listas de chequeo de los niveles de verificación ASVS.

Actividades

* Identificar que herramientas y talento humano requieren para completar la lista de chequeo de los niveles de verificación en ASVS.
* Generar los artefactos de salida a partir del análisis realizado en las actividades anteriores.

## Determinación del estado deseado de la Seguridad en Aplicaciones Web

En esta fase del plan estratégico se establece el nivel de verificación al que se aspira llegar en cuanto a seguridad de las aplicaciones web. Teniendo presente las listas de chequeo recopiladas en el paso previo, los objetivos definidos plenamente por las partes interesadas, las limitaciones a las que se enfrenta la organización y el análisis del estado actual en materia de seguridad en aplicaciones web. A continuación, se presentan los siguientes artefactos de entrada, salida y las actividades en esta fase:

1. **Artefactos de entrada**

* Listas de chequeo fase anterior.
* Objetivos del plan estratégico de la seguridad en aplicaciones web.
* Limitaciones de negocio o de recursos de TI que imposibiliten llegar al nivel de verificación deseado.

Actividades

* Recopilar datos de otros sistemas para realizar procesos comparativos.
* Mesas de trabajo para validar las listas de chequeo de la fase anterior.

1. **Artefactos de salida**

* Meta establecida del nivel de verificación ASVS.

Actividades

* Generar los artefactos de salida a partir del análisis realizado en las actividades anteriores.

## Definición de los Indicadores en Aplicaciones Web

Esta fase es vital en el plan estratégico de seguridad en aplicaciones web debido a que le proporciona a los interesados la manera correcta de cuantificar el rendimiento real del sistema después de realizar un plan estratégico de seguridad en aplicaciones web; permitiéndole a la gerencia un punto de referencia en la toma de decisiones, en las oportunidades de mejora o en el proceso de pivotar la dirección de los objetivos trazados por la corporación para lograr ventajas competitivas.

Dentro de esta etapa se definen cuatro importantes actividades que al final permiten gestar un paquete de métricas para implementar. Para este paso se establecen los siguientes artefactos de entrada, salida y las actividades:

1. **Artefactos de entrada**

* Meta establecida del nivel de verificación ASVS.
* Objetivos del plan estratégico de la seguridad en aplicaciones web.

Actividades

* Recopilar las mejores prácticas de expertos en la temática.
* Realizar mesas de trabajo que permitan obtener una lista de chequeo con los elementos que generen valor agregado al cliente de las aplicaciones.
* Evaluar los indicadores y métricas existentes contra la lista de valor agregado e identificar las falencias, lagunas y hallazgos.

1. **Artefactos de salida**

* Análisis de brechas encontradas.
* Listado del estado actual de los indicadores y métricas.
* Paquete final de indicadores y métricas para el plan estratégico de seguridad de las aplicaciones web.

Actividades

* Realizar pruebas “benchmark” al sistema para analizar las brechas en materia de indicadores y métricas.
* Desarrollar un diccionario de indicadores y métricas que detalle la definición y el cálculo de la misma.
* Validar y asegurar que los indicadores y métricas preliminares se alinean con el objetivo del plan estratégico de la seguridad en aplicaciones web para su posterior aprobación por todos los “stakeholders”.
* Generar los artefactos de salida a partir del análisis realizado en las actividades anteriores.

Ahora:

* **Lista de chequeo requisitos de seguridad en aplicaciones web.** Para conseguir los indicadores y sus métricas es necesario conocer los requisitos de seguridad que se pretenden verificar en una aplicación web, es por ello que se presenta a continuación una lista de estos, basado en la propuesta del ASVS en su versión 4.0. (OWASP. 2021).
* **Arquitectura, diseño y modelado de amenazas.** Se coloca al inicio de todos los requisitos por analizar debido al papel fundamental que juega el diseño, el modelado y la arquitectura para cualquier sistema “software”, puesto que, este define la estructuras que componen la aplicación y la forma como se comunican entre ellas, para tener la capacidad de satisfacer los atributos de calidad del sistema. Es por ello que para este elemento se debe asegurar que se cumplan los requisitos que se observan en el PDF **Lista de chequeo: arquitectura, diseño y modelado de amenazas**, el cual se encuentra en la carpeta Anexos.
* **Autenticación.** El siguiente requisito en ser verificado es el mecanismo de inicio y control del estado de la sesión de un usuario a través del método de autenticación. Este proceso es el punto de entrada a la aplicación en consecuencia asegurar este componente es la principal tarea del equipo de TI. Por este motivo, para la autenticación se deben controlar los ítems que se presentan en el PDF **Lista de chequeo: autenticación**, el cual se encuentra en la carpeta Anexos.
* **Control de acceso.** Este requisito consiste en el mecanismo que autoriza el acceso y uso de los recursos acorde con los permisos del solicitante. Teniendo en cuenta esto, es obligatorio que se validen cada uno de los ítems que se presentan en el PDF **Lista de chequeo: control de acceso**, el cual se encuentra en la carpeta Anexos.
* **Manejo de entrada de datos maliciosos.** Otro requisito sensible en la seguridad de las aplicaciones web consiste en la validación de entrada de datos maliciosos al sistema provenientes de ataques oportunistas desde el mismo usuario de la aplicación. Esta debilidad permite la ejecución de distintos ataques al sistema de ficheros, ataques como inyección de intérprete, “cross-site scripting” (XSS), ataques “locale/Unicode”, inyección de SQL y desbordamientos de búferes. Por lo cual es necesario validar que para este elemento se validen cada uno de los ítems que se presentan en el PDF **Lista de chequeo: manejo de entrada de datos maliciosos**, el cual se encuentra en la carpeta Anexos.
* **Criptografía en el almacenamiento.** Esta verificación hace referencia a la importancia del encriptamiento de los datos más sensible dentro de una aplicación; es por ello que el sistema debe asegurar que se cumplan los ítems que se presentan en el PDF **Lista de chequeo: criptografía en el almacenamiento**, el cual se encuentra en la carpeta Anexos.
* **Gestión y registro de errores.** La gestión y registro de errores en “log”, tiene como objetivo proporcionar un mecanismo reactivo de respuesta a incidentes para los usuarios, administradores y equipos de TI, creando registros de “log” bastante significativos para los desarrolladores proporcionando datos útiles y sin agregar ruido innecesario. Los datos almacenados en el “log” deben presentar una cobertura acorde a las reglamentaciones y leyes dispuestas por el país ante los temas de privacidad de la información o directivas similares y se debe garantizar que se gestiona de forma segura. Es por ello que la aplicación debe contar con la validación de los ítems que se presentan en el PDF **Lista de chequeo: gestión y registro de errores**, el cual se encuentra en la carpeta Anexos.
* **Protección de datos.** Este requisito de verificación se centra en la confidencialidad, integridad y disponibilidad de los datos, teniendo presente que esto se aplica en un servidor de aplicaciones, lo que permite asumir que posee protecciones confiables y suficientes. Sin embargo, un sistema debe comprender que todos los dispositivos de los usuarios pueden comprometer la seguridad de alguna forma en el proceso de transmisión de información, por lo que es responsabilidad de la aplicación que los datos persistidos en algún dispositivo esté cifrado y sea complicado o imposible obtenerlo, alterarlo o divulgarlo. A continuación, en el PDF **Lista de chequeo: protección de datos**, el cual se encuentra en la carpeta Anexos, se identifican los pormenores que se debe cumplir en temas de protección de datos.
* **Seguridad de las comunicaciones.** Para el elemento de seguridad de las comunicaciones es obligatorio que se cubran los requisitos que se presentan en el PDF **Lista de chequeo: seguridad de las comunicaciones**, el cual se encuentra en la carpeta Anexos.
* **Configuración de seguridad HTTP.** La configuración de seguridad HTTP necesita asegurar los ítems que se presentan en el PDF **Lista de chequeo: configuración de seguridad HTTP**, el cual se encuentra en la carpeta Anexos.
* **Controles maliciosos.** El requisito de control malicioso es susceptible de no ser detectado con facilidad, puesto que debe ser revisado minuciosamente el código fuente por un experto en el tema de forma manual para detectar bombas lógicas alojadas en alguna línea indetectable a simple vista. Por ello, en este se deben satisfacer los elementos de alto nivel presentados en el PDF **Lista de chequeo: controles maliciosos**, el cual se encuentra en la carpeta Anexos.
* **Lógica de negocios.** Para este requisito de verificación se necesitan satisfacer las características que se encuentran en el PDF **Lista de chequeo: lógica de negocios**, el cual se encuentra en la carpeta Anexos.
* **Archivos y recursos.** Para los archivos y recursos se deben remediar los requisitos que se encuentran en el PDF **Lista de chequeo: archivos y recursos**, el cual se encuentra en la carpeta Anexos.
* **Servicios web.** Es necesario que la aplicación verifique que al utilizar servicios web “REST” o “SOAP”, se mantengan los requisitos que se encuentran en el PDF **Lista de chequeo: servicios web**, el cual se encuentra en la carpeta Anexos.

A partir de los anteriores requisitos, el equipo TI puede plantear la selección de ciertos KPI y métricas que luego serán validadas y aprobadas por los interesados; que le permitan posteriormente a la gerencia identificar si el plan estratégico de la seguridad en aplicaciones web ha impactado positivamente los procesos y los sistemas aportando a los objetivos del negocio.

### Indicadores Claves de Rendimiento (KPI, “Key Performance Indicator”)

Para afirmar que un proceso, servicio o producto mejora durante la ejecución de un plan, es necesario que existan mediciones que permitan evaluar, comparar y contrastar resultados a fin de respaldar esta declaración (G. V. Ríos, John A. Bohada e I. A. Delgado, 2018). Por esta razón, el área de TI durante la fase de “Definición de los Indicadores y Métricas de la Seguridad en Aplicaciones Web”, del plan estratégico, debe centrar sus esfuerzos en identificar los KPI y las métricas asociadas a ellos para brindar a los “stakeholders” un conjunto de herramientas e información puntual que les permitirá tomar decisiones y realizar acciones proactivas/reactivas frente a los objetivos del negocio y el direccionamiento estratégico de la organización.

Ahora bien, antes de identificarlos es importante entender y diferenciar un KPI de una métrica, para que el proceso de establecer ambos elementos sea exitoso:

* **Métrica.** Se define como cualquier concepto que pueda valorarse cuantitativamente; por ejemplo, el número de intentos de acceso a una aplicación o el número de transacciones completadas, etc. Es decir, la métrica representa el dato primitivo o valor atómico medido como respuesta a una pregunta concreta.
* **KPI.** Está relacionado con el éxito de cada uno de los objetivos propuestos por una organización. En concreto, un indicador se alinea al plan propuesto por la alta gerencia, por ejemplo, la tasa de crecimiento de los ingresos, el porcentaje de disponibilidad de la aplicación, o el tiempo de respuesta después de un fallo, entre otros.

Para establecer los KPI para el plan estratégico de seguridad en aplicaciones web que se van a estar monitoreando, se pueden seguir algunos consejos simples que lo facilitan:

1. **Seleccionar del listado de requisitos por componente los criterios básicos a medir basados en el nivel de verificación en el plan estratégico en la fase tres.** Tener presente que los KPI deben ser muy concretos y específicos, apuntando a un solo objetivo y midiendo un solo parámetro y sin tener múltiples aristas.

Por ejemplo:

**Requisito de verificación de Arquitectura, Diseño y Modelado de amenazas.**

* Identificar información sensible de código que es visible del lado del cliente.
* Contar el número de llamadas a librerías externas o programas de terceros, potencialmente inseguros.

1. **Seleccionar el requisito y la medida a realizar.** Después de seleccionado el requisito y la medida a realizar se debe validar contra qué factor se va a comparar.

Por ejemplo:

* Requisito de verificación de arquitectura, diseño y modelado de amenazas.
* Disminuir a cero la información sensible de código visible del lado del cliente.
* Llevar el código a solo realizar un 10 % de llamadas a librerías externas o programas de terceros.

1. **Tener la capacidad de decisión con respecto a los resultados positivos o negativos.** Por último, es necesario que el equipo de TI tenga la capacidad de decisión con respecto a los resultados positivos o negativos en el tiempo, puesto que, los KPI se usan para plantear estrategias que puedan mitigar problemas que ocurren o que podrían suceder.

No basta solo con medir, sino que se debe tener plena confianza en las oportunidades de mejora que brindan los indicadores y sus respectivas métricas.

## Definición de Métricas de Seguridad en Aplicaciones Web

Como se indicó anteriormente, los KPI necesitan de ciertas métricas para su confección y refinamiento; sin embargo, está claro que se puede seleccionar cualquier variable de la gran cantidad que pueden generar un programa sobre el ecosistema de temas en seguridad; por esta razón, es pertinente aclarar cómo se debe establecer si una métrica está acorde a lo que se necesita para el plan estratégico de seguridad de las aplicaciones web. Así, las características más sobresalientes de las métricas son:

* **Dependencia del proceso.** Estas deben ser derivadas específicamente del resultado del proceso para garantizar que tengan un impacto significativo y sean fácilmente evaluables.
* **Confiable.** Si los datos que puede tomar la métrica no son precisos ni verificables, se dice que son aleatorios y arbitrarios, por tanto, no debería plantearse como una métrica a tener en cuenta.
* **Cuantificable.** Estas deben ser concretas y cuantificables, con una facilidad para expresarse en unidades relevantes y nunca deben ser ponderadas en conceptos como “bueno” o “bonito”, etc.
* **Alineación a los objetivos.** La información que proporcione una métrica debe apoyar el cumplimiento de los objetivos de la organización.
* **En curso y comparable.** La métrica debe proporcionar un valor que permita su comparación en función del tiempo y no en indicadores de rendimiento de una sola ocasión.

Si las métricas seleccionadas poseen estas características, permitirán a la corporación comprender de mejor forma los riesgos a los que se pueden enfrentar, las amenazas y problemas emergentes, identificar y aislar las debilidades y tener una evaluación más acertada del desempeño de los controles implementados en materia de seguridad en aplicaciones web; del mismo modo, se crearán oportunidades de mejora y actualización en procesos, herramientas y tecnologías en comparación a los actuales evidenciando el incremento y evolución de la cultura organizacional en materia de seguridad en sistemas. Las métricas en la temática de ciberseguridad pueden clasificarse de la siguiente manera:

* **Métricas en procesos de seguridad.** Estas métricas tienen como objetivo medir los procesos y procedimientos implicados en las políticas de seguridad de la organización, buscando que los indicadores se relacionen con el nivel de seguridad a través del abordaje de las normas y estándares del gobierno en seguridad.
* **Métricas de red.** Las métricas de red están dirigidas o impulsadas por productos tales como “firewalls”, sistemas de detección de intrusos, etc. Las cuales están ampliamente disponibles y utilizadas para conseguir los objetivos y brindar a la organización la sensación de control, inclusive teniendo interfaces de usuarios con gráficos que permiten una lectura sencilla.
* **Métricas de “software”.** Las métricas que tienen como finalidad el “software” tienen una dependencia directa del ciclo de vida de la construcción de aplicaciones y la arquitectura, el diseño y el modelado del “software”. Las organizaciones deben documentar sus métricas en un formato estándar para asegurar la aplicación del modelo, adaptación, recopilación y presentación de informes.

# Ejecución del Plan Estratégico de la Seguridad en Aplicaciones Web

La última fase del plan estratégico de la seguridad consiste en su ejecución. Para ello se tiene los siguientes artefactos de entrada, salida y las actividades en esta fase:

1. **Artefactos de entrada**

* Paquete final de indicadores y métricas para el plan estratégico de seguridad de las aplicaciones web.
* Objetivos del plan estratégico de seguridad en aplicaciones web.
* Meta establecida del nivel de verificación ASVS.

1. **Actividades**

* Estimar los recursos requeridos y los costos para la implementación por cada una de las acciones definidas en el plan estratégico.
* Definir un programa continuo de sensibilización y capacitación en seguridad de aplicaciones web, que facilite al equipo de TI la adopción del plan estratégico de manera eficaz.
* Establecer planes de acción y futuros proyectos para tratar de minimizar la brecha y pensar en llegar a un estado superior de nivel de verificación en seguridad de aplicaciones web.
* Generar los artefactos de salida a partir del análisis realizado en las actividades anteriores.

1. **Herramientas de calidad de código**

* Hoja de ruta correspondiente al plan de acción para la implementación de la seguridad en aplicaciones web, en la cual se priorizan las iniciativas que se deben llevar a cabo para cerrar la brecha y alcanzar el estado deseado.
* Plan de uso y apropiación de la estrategia.
* Paquete final de indicadores y métricas para el plan estratégico de seguridad de las aplicaciones web.

A recordar:

Los indicadores claves de rendimiento, las métricas, los niveles de verificación de ASVS y los requisitos de verificación detallada para la seguridad en aplicaciones web, permiten construir un plan estratégico de seguridad que conduce a una organización a la apropiación y ejecución de una hoja de ruta en su intención de lograr que las aplicaciones que hacen parte de su ecosistema y el núcleo del funcionamiento del negocio que se encuentren en el nivel de verificación adecuado según los estándares recomendados por los marcos de trabajo mencionados en el OWASP y el ASVS para su sector dentro de la industria; permitiendo que estos sistemas puedan escalar en un futuro y responder ante los constantes cambios que presenta el “software” en el mundo actual.

El plan estratégico propuesto será para las compañías una herramienta de gran utilidad para prevenir atacantes y vulnerabilidades asociadas a las aplicaciones web, gracias a que la implementación de este plan permitirá a los equipos de desarrollo tomar decisiones oportunas ante eventualidades y de esta forma perfeccionar gradualmente los mecanismos de seguridad y de control de los sistemas de las organizaciones.

En el siguiente capítulo se proponen algunas herramientas que brindan al equipo de desarrollo opciones para el monitoreo automatizado de las aplicaciones en temas de seguridad web.

## Herramientas de Monitoreo de Seguridad en Aplicaciones Web

En el capítulo anterior se introdujo el estándar de verificación en seguridad en aplicaciones ASVS, el cual puede convertirse para las organizaciones en una hoja de ruta para la verificación de sus sistemas en conjunto con los proyectos abiertos y sin restricciones aprovechándose de su documentación, su código fuente y el manejo de la autenticación al sistema, particularmente para los niveles de verificación estándar y avanzado.

En la gran mayoría de ejercicios de revisión de código y test de penetración, todos los hallazgos presentados en el informe que recibe la dirección general se componen exclusivamente de defectos de seguridad; no obstante, es necesario incluir cuál es el alcance de la verificación, el resumen detallado de los resultados del nivel de verificación anexando a este las pruebas que se completaron, las pruebas exitosas y las fallidas, estableciendo además una posible solución a las erradas.

Es por ello que las organizaciones promueven prácticas que mantengan la documentación asociada a las pruebas en formato electrónico, usando imágenes, videos, registros “logs”, o “scripts” entre otros, puesto que, esta serie de archivos sirven de hallazgos para los desarrolladores cuando tengan dudas en sus procesos.

La comunidad OWASP en su afán de perfeccionar estos procesos han planteado una serie de herramientas automatizadas que intentan brindar una gran cobertura y ejecución de gran cantidad de parámetros a los equipos de desarrollo con múltiples formas de ataques y entradas maliciosas. No obstante, ASVS indica que no es suficiente con solo ejecutar una herramienta y confiar en un único informe de fallas, puesto que, esta sola herramienta no proporciona la evidencia necesaria para abordar todos los problemas de un nivel de verificación para indicar que está comprobado completamente.

En este caso se debe tener presente que la industria “software” madura a un ritmo acelerado y en materia de seguridad en aplicaciones la diferencia entre pruebas manuales y asistidas por herramientas es significativamente mínima. Es por ello, que es importante aclarar que estas herramientas automatizadas simplemente se adaptan para realidades puntuales y no tienen la capacidad de cubrir todos los requerimientos que a día de hoy existen.

A continuación, se presentan algunos proyectos de OWASP que utilizan ASVS y proveen una herramienta para los equipos de desarrollo.

## El marco de conocimiento de seguridad (“Security Knowledge Framework”)

La experiencia de la comunidad de OWASP permite identificar que el nivel de seguridad en las aplicaciones web no es significativamente alto para garantizar tranquilidad a las organizaciones. Esto debido a que muchos equipos de desarrollo no dimensionan el riesgo y las vulnerabilidades que pueden ser explotados por los piratas informáticos. Razón por la cual, se decantan por desarrollar un “framework” utilizando ASVS como guía para crear un sistema que sirva como hoja de ruta desde el primer paso para el desarrollo de aplicaciones seguras.

El OWASP-SKF es una aplicación web de código abierto escrita en el lenguaje de programación Python usando el “framework” Flask, que le proporciona a los desarrolladores de “software” una potente herramienta para que aprendan los principios de codificación segura en múltiples lenguajes de programación; entrenándolos en la escritura de código seguro desde la concepción, diseño e implementación de un sistema. La premisa es plantar la semilla de la seguridad en el desarrollador desde el diseño hasta la última etapa del ciclo de vida, a través de una serie de listas de verificación, como por ejemplo OWASP- ASVS, OWASP-MASVS y la ejecución de laboratorios y prácticas que permiten explorar los niveles de verificación de seguridad (“Security Knowledge Framework”, 2022).

### OWASP ZAP (“Zed Attack Proxy”)

ZAP es una herramienta de código abierto de fácil uso, flexible, extensible y gratuita mantenida por el esfuerzo conjunto del OWASP, que permite realizar pruebas de penetración para aplicaciones web. La aplicación actúa como un “proxy” ubicándose entre el navegador y el sistema a testear para la búsqueda de vulnerabilidades; con el fin de interceptar e inspeccionar los mensajes enviados entre estos dos y con la disponibilidad de realizar cambios en el contenido de este y reenviarlo al destino.

Además ofrecen la opción de escaneo automatizado para una gran variedad de niveles de verificación ASVS y contiene otras herramientas para encontrar vulnerabilidades de seguridad manualmente. Esta aplicación ofrece al público versiones para los distintos sistemas operativos y también se puede usar a través de “docker”; otras opciones adicionales y complementos son posible descargarlos gratuitamente desde el ZAP “Marketplace”. Por último, la herramienta permite a los desarrolladores usarla como aplicación independiente o como un proceso que corre en “background” en el sistema operativo. (OWASP “Zed Attack Proxy”, 2022).

Otro punto importante que vale la pena resaltar es que este proyecto al ser de código abierto permite a los desarrolladores examinarlo sin inconvenientes; dándoles la posibilidad de identificar cómo se implementa alguna funcionalidad. Además ZAP, brinda la opción de trabajar en su proyecto en la corrección de errores, aditamento de funciones o funcionalidades, o creación de complementos para situaciones especiales requeridas.

### OWASP “Juice Shop”

Este proyecto de OWASP consiste en la exposición de una aplicación web moderna y sofisticada que usa tecnologías actuales como Node.js, Express y Angular, pero que presenta una gran cantidad de vulnerabilidades en materia de seguridad; esto con el fin de realizar capacitación en esta materia, demostraciones de fallos y concientización para la comunidad “software” en general. Además sirve también como ejemplo para las otras herramientas como OWASP ZAP, entre otras.

“Juice Shop” contiene todas las vulnerabilidades del OWASP “Top Ten”, junto a otro gran número de falencias en temas de seguridad que se basan en todos los hallazgos encontrados en aplicaciones web en la comunidad global. Dentro de ella se pueden descubrir gran variedad de ejemplos y desafíos de piratería informática de dificultad variable que el usuario debe explotar; a medida que se avanza con estas dinámicas el progreso se aprecia en un tablero de puntuación.

Como se mencionó anteriormente esta herramienta le puede ayudar a los expertos en el área de seguridad y a los equipos de “software” como aplicación para adiestrarse y comprobar qué tan bien sus herramientas se adaptan a las interfaces de aplicaciones con gran cantidad de código JavaScript y API REST. (OWASP “Juice Shop” 2022).

### Demos herramientas “Juice Shop” y OWASP ZAP

A continuación, en el siguiente video se presenta una demostración del uso de estas aplicaciones. Antes de empezar con las herramientas es necesario instalar algunos paquetes en el sistema referenciados en este documento (NodeJs, 2022) y (Git, 2022).

1. Herramientas de seguridad



[**Enlace de reproducción del video**](https://www.youtube.com/watch?v=3Ul1dN-0BfE)

|  |
| --- |
| **Síntesis del video: Herramientas de seguridad** |
| Tutorial que presenta las herramientas que se utilizan para seguridad, y cómo utilizarlas. |

Es importante tener presente que en el mercado existe una gran variedad de estas tanto libres y gratuitas como pagadas y de terceros; todas con características especiales que se adaptan a las necesidades de los expertos y las condiciones de las vulnerabilidades a medir dentro de las aplicaciones web.

## Medición de tráfico de red

Actualmente es casi imposible para la sociedad pensar en su día a día sin la conexión a internet. La red mundial creció en las últimas décadas a un ritmo increíblemente rápido, lo que produjo un aumento sustancial en el tráfico de datos e información en la red, proveniente en gran medida de los servicios de transmisión, las redes sociales y las aplicaciones web en general (Pries, R, 2009).

Con este panorama las redes deben soportar un aumento en el promedio de carga de tráfico y un cambio errático en los patrones de los datos. Por tanto, las organizaciones y sus equipos de desarrollo necesitan contar con una supervisión y análisis efectivo de todo el tránsito de datos de información; debido a que detectar eventos externos, cuellos de botella, patrones erráticos, actividades maliciosas y/o procesos poco usuales en la red, permite anticiparse de manera oportuna y efectiva a potenciales problemas cuando ocurren o antes de que sucedan. De modo que, los servicios y/o aplicaciones no presenten interrupciones, indisponibilidad, intermitencia o suspensión durante ráfagas o durante largos periodos de tiempo (Xinyu, X., S. Mishra, 2009).

La medición de tráfico es una técnica que supervisa y notifica de manera constante el flujo de datos de la red al equipo de gestión y administración de las aplicaciones y/o servicios cada vez que ocurre un evento. Para realizar dicho monitoreo es posible encontrar en el mercado distintas herramientas que utilizan múltiples técnicas para analizar el tráfico total.

A continuación, se exponen algunos enfoques de monitoreo de red y se presentan algunas herramientas para dicha tarea:

### Wireshark

Este proyecto inició su desarrollo en el año 2007 y continúa aún en vigencia exhibiendo una librería de información amplia y con múltiples protocolos de red, permitiendo su ejecución en diferentes sistemas operativos nativos, tales como Linux, MacOs o Windows (Dabir, A., A. Matrawy, 2007).

Wireshark captura todos los paquetes del flujo de la red para luego exponerlos completamente en una interfaz gráfica de usuario con metainformación de cada dato analizado, brindándole a los administradores la oportunidad de identificar qué equipo o dispositivo está tratando de interactuar con otro, ofreciendo, además, corroborar si un paquete tiene una estructura correcta o no (Shaoqiang, W., X. DongSheng, y Y. ShiLiang, 2010).

Esta herramienta posee una función de línea de comandos que recopila paquetes de transmisión, que posteriormente son archivados en disco para un análisis y presentación en la interfaz mencionada anteriormente. La aplicación debe ser ejecutada con un usuario administrador o en su defecto con privilegios, o en consecuencia no será capaz de visualizar las interfaces de red disponibles para análisis y monitoreo.

La herramienta en la interfaz provee una opción de filtrado por las distintas columnas que presenta directamente; como por ejemplo fuente, destino, protocolo, etc. Es importante detener la captura de paquetes, puesto que si no se realiza Wireshark seguirá analizando el tráfico en modo silencioso, incurriendo en consumo de memoria del sistema.

Si se hace clic en cualquier paquete, se mostrará información detallada sobre el paquete capturado en la mitad inferior de vista.

### NTOP

Esta herramienta ofrece una solución para monitorear el tráfico con un nivel sustancialmente alto de visibilidad de los datos, permitiendo realizar un análisis detallado de los protocolos de red y los paquetes transmitidos.

A continuación, se podrá conocer detalles de esta herramienta:

* **Tablero NTOP.** El tablero NTOP provee al administrador la oportunidad de conocer qué está sucediendo con el flujo de información entre dos dispositivos a través de los puertos TCP o UDP. Esta imagen no es estática, sino que se actualiza cada cierto espacio de tiempo especificado por NTOP; mostrando los diferentes dispositivos conectados a la red y la cantidad de datos que recibe de otros dispositivos en particular, representando cada tipo de tráfico con un color diferente.
* **Segregación en términos de su tipo y sus dimensiones.** Además de las segregaciones, también posee una librería que brinda un proceso de inspección profundo del flujo de información, a través de la captura, análisis y flujos de paquetes, como, por ejemplo, obtener en un marco de tiempo de hasta un año los resultados de un protocolo en particular.
* **Librería “libcap” de NOTP.** NTOP usa la librería “libcap” que permite la captura de paquetes en distintos sistemas operativos, logrando que su código fuente sea el mismo haciendo que la herramienta sea portable. Además, el almacenamiento en búfer de los paquetes reduce la pérdida en caso de ráfagas de tráfico lo que brinda fiabilidad al momento de analizar los datos; los cuales son procesados una a la vez. La metainformación de los dispositivos se almacena en las tablas “hash” como datos de entrada y salida ordenados acorde con los protocolos de red; por último, NTOP presenta una gran cantidad de filtros almacenados y se aplican directamente al tráfico capturado.
* **Lista de “host” detectados por el “software”.** Esta imagen muestra la lista de “host” detectados por el “software”, los cuales son reconocidos cada vez que reciben tráfico proveniente de algún otro dispositivo.

NTOP tiene la capacidad de decodificar los paquetes HTTP, por lo que logrará identificar el sistema operativo del servidor en el cual se está ejecutando. Además es posible apreciar el desglose del tráfico en términos de datos entrantes/salientes y la aplicación proporciona la descarga de un objeto JSON con estos datos, para que se pueda manipular o almacenar en una base de datos.

### Microsoft Message Analyzer

Esta herramienta creada para el sistema operativo Microsoft contiene entre sus características principales el rastreo en vivo (“Live Trace”) que logra capturar y rastrear el flujo de tráfico de los dispositivos, HTTPS sin cifrar, el tráfico de la capa de enlace local de NDIS, las interfaces remotas de red, el tráfico IPSEC sin cifrado, VPN y acceso directo a red.

**Características de Microsoft Message Analyzer**

Permite la captura del tráfico de máquinas remotas usando para ello una sesión desde el menú, realizando una transferencia de los datos de dicha máquina al dispositivo local, para realizar un análisis a los registros de texto y paquetes.

**Nivel de trabajo**

Utiliza el flujo a un alto nivel de trabajo presentando los mensajes de varios archivos de registro, ordenados y clasificados por tiempo, basado en la información de la sesión en curso, brindando detalles sobre dicho mensaje a través de la ventana de herramientas.

**Organización de registros**

Los registros pueden ser organizados por sus propiedades tales como módulo, fuente y destino. La vista de los datos muestra texto, imágenes y permite verlos también en formato xml. Cuando se inicia la herramienta, esta permite seleccionar una sesión de todas las detectadas en la lista, presentando el número total de mensajes entrantes y proporcionando una funcionalidad que abre varias ventanas de análisis a la vez acoplables entre ellas.

**Aplicación de Microsoft**

Permite a los usuarios una serie de filtros que le permiten aislar y centrarse en datos específicos para su posterior análisis. Algunos de los filtros que posee son:

* Filtro rápido.
* Palabras clave.
* Filtro de sesión.
* Filtro de vista.

Cada uno de estos realiza una tarea única y la herramienta brinda al usuario la opción de descifrar datos con los protocolos “Transport Layer Security” (TLS) y “Secure Sockets Layer” (SSL).

### “Paessler Router Traffic Grapher” (PRTG)

Esta herramienta de análisis y rastreo de red que identifica el tráfico que más congestiona el flujo de datos permite prevenir eventos y problemas que se generen por consecuencia de ello. PRTG provee características avanzadas en el descubrimiento automatizado de la red, con un énfasis en la seguridad ya que alerta al usuario ante eventuales fallos o emergencias a través del envío de correo o mensaje de texto.

La aplicación tiene compatibilidad con navegadores, interfaces de usuario de Windows y iPhone, permitiendo el acceso en cualquier momento y desde cualquier ubicación. Además, le brinda al usuario el monitoreo del ancho de banda de la red y los clientes conectados a la misma, con el plus de indicar con qué propósito se conectan (PRTG, 2022).

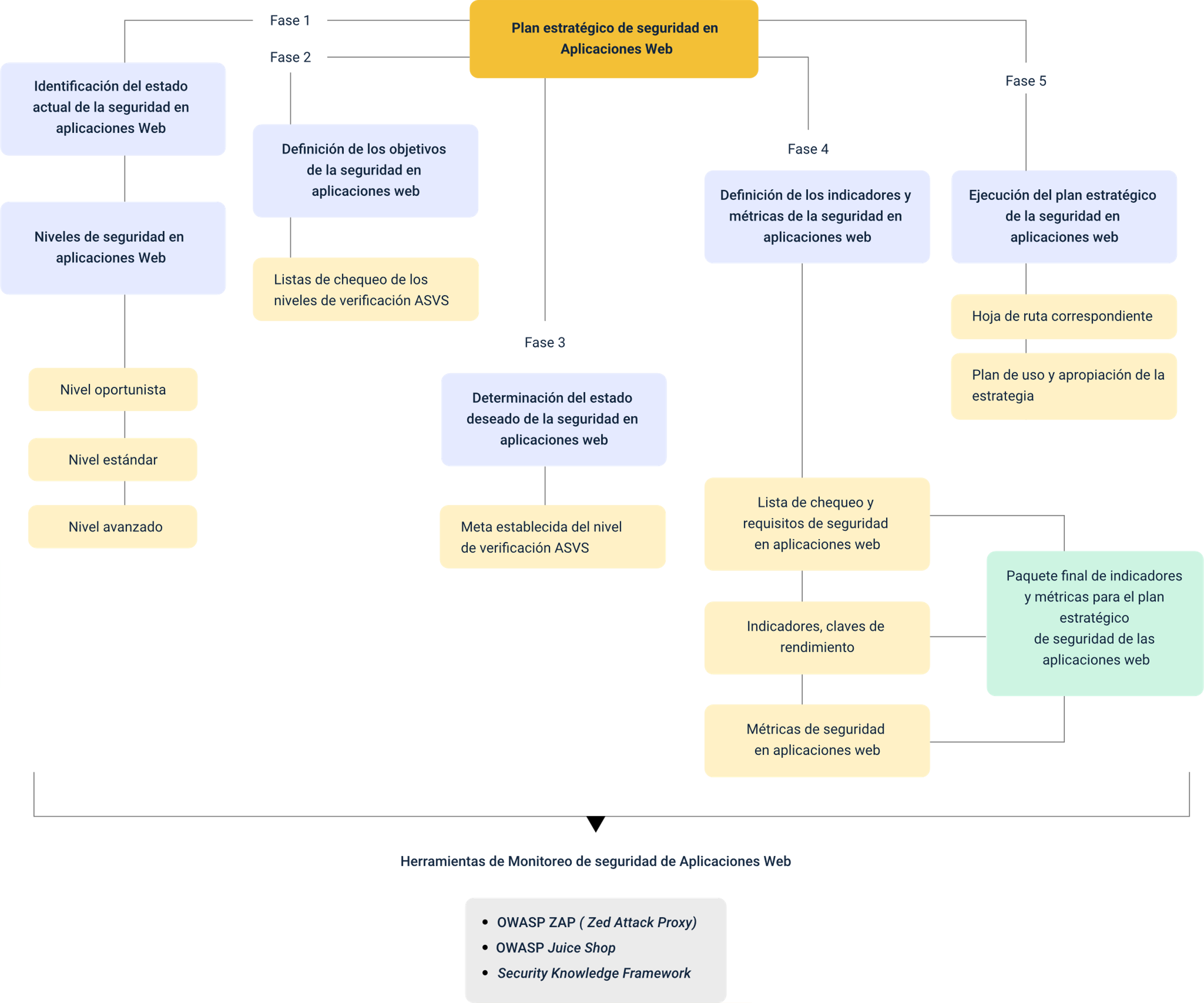
PRTG posee más de 200 tipos de sensores para todos los servicios de red comunes, incluidos HTTP, SMTP/POP3 y FTP; admitiendo diferentes protocolos de red para la recolección de la información, tales como: SNMP y WMI, Packet Sniffing, NetFlow, IPFIX, JFlow y SFlow. A continuación, se podrá conocer las características de esta herramienta:

* **Interfaz de usuario accedida desde la web.** En la imagen se precisa la interfaz de usuario accedida desde la web con gráficos de datos en tiempo real; esta interfaz es alimentada por los sensores HTTP que constantemente monitorean el tiempo de respuesta, sino hay respuesta o se recibe un estado 404, dicho sensor entra en estado de inactividad. La herramienta presenta los “bytes” de entrada, el tiempo de descarga, la velocidad y el tiempo del primer paquete entrante para cada uno de los sensores; con lo cual, el administrador puede establecer ciertos parámetros a modo de umbral que disparan las alarmas previstas. La herramienta presenta al usuario la posibilidad de descargar informes detallados en formato HTML o en PDF acorde a los parámetros seleccionados.
* **Sensores a partir de transacciones.** Esta imagen presenta algunos de los sensores que ofrece la herramienta permitiendo a los administradores de la red simular una transacción completa en su aplicación web mediante una serie de hasta 10 enlaces que ayudan a comprobar la accesibilidad y el contenido específico de cada una de estas páginas. PRTG ofrece a los usuarios analizar el flujo completo de una aplicación ejecutada en un servidor incluyendo servicios específicos como: monitoreo pasivo de sensores, rendimiento y tiempo de conexión TCP entre otros (PRTG, 2022).

En esta sección se presentan algunas herramientas que permiten analizar y monitorear el tráfico y el flujo de red, puesto que la carga en promedio de las redes ha aumentado con el paso del tiempo, convirtiendo sus patrones y comportamiento en impredecibles para los administradores; por lo cual, se hace necesario supervisar y rastrear el tráfico para solucionar y resolver problemas de manera efectiva cuando suceden o antes de que puedan ocurrir, de modo que los servicios y aplicaciones no se detengan por largos períodos de tiempo.

Síntesis

En el siguiente recurso podrá evidenciar un resumen de este componente formativo:



El esquema presenta la síntesis de la temática estudiada en el componente formativo, comenzando por el plan estratégico de seguridad en aplicaciones web, el cual está compuesto por:

* Fase 1: identificación del estado actual de la seguridad en aplicaciones web: compuesto por los niveles de seguridad en aplicaciones web: nivel oportunista, nivel estándar y nivel avanzado.
* Fase 2: Definición de los objetivos de la seguridad en aplicaciones web: compuesto por las listas de chequeo de los niveles de verificación ASVS.
* Fase 3: Determinación del estado deseado de la seguridad en aplicaciones web: compuesto por la meta establecida del nivel de verificación ASVS.
* Fase 4: Definición de los indicadores y métricas de la seguridad en aplicaciones web: compuesto por la lista de chequeo y requisitos de seguridad en aplicaciones web, indicadores claves de rendimiento y métricas de seguridad en aplicaciones web, los cuales definen el paquete final de indicadores y métricas para el plan estratégico de seguridad de las aplicaciones web.
* Fase 5: Ejecución del plan estratégico de la seguridad en aplicaciones web: compuesto por la hoja de ruta correspondiente y el plan de uso y apropiación de la estrategia.

Material complementario

|  |  |  |  |
| --- | --- | --- | --- |
| Tema | Referencia | Tipo de material | Enlace del recurso |
| Definición de Métricas de Seguridad en Aplicaciones Web | OWASP Security Qualitative Metrics. (s/f). Owasp.Org. | Página web | <https://owasp.org/www-project-security-culture/v10/8-Metrics/> |
| El marco de conocimiento de seguridad (“Security Knowledge Framework”) | wstg: The Web Security Testing Guide is a comprehensive Open Source guide to testing the security of web applications and web services. (s/f). | Página web | <https://github.com/OWASP/wstg> |
| El marco de conocimiento de seguridad (“Security Knowledge Framework”) | Pereira, C. P. T.-D. [DavidPereira]. (2021, marzo 4). Curso de Pentesting a Aplicaciones WEB (Video 1) OWASP Juice Shop. Youtube. | Video | <https://www.youtube.com/watch?v=6eAJ1dinee0&t=690s> |
| El marco de conocimiento de seguridad (“Security Knowledge Framework”) | Tech, T. [TaggartTech]. (2021, septiembre 8). #AttackOnTuesday | OWASP ZAP/Juice Shop. Youtube. | Video | <https://www.youtube.com/watch?v=4CNWTkAbT4o&t=801s> |
| El marco de conocimiento de seguridad (“Security Knowledge Framework”) | UskoKruM [UskoKruM2010]. (2021, agosto 19). Cómo Descargar e Instalar Node.js en Windows 10 | Explicación Sencilla. Youtube. | Video | <https://www.youtube.com/watch?v=ipmhBYqIP44&t=4s> |
| El marco de conocimiento de seguridad (“Security Knowledge Framework”) | Eber, D. [donEber]. (2021, junio 17). Instalación de Git en Windows paso a paso | [2021| 2022]. Youtube. | Video | <https://www.youtube.com/watch?v=cYLapo1FFmA&t=2s> |

Glosario

**API “Endpoints”:** son las URLs de un API o un “back-end” que responden a una petición.

**API “REST”:** es una arquitectura que permite la interacción con los servicios web de RESTful.

**Aplicaciones POS:** son un tipo de programas destinados a sistematizar el control de ventas e inventarios del punto físico de un negocio.

**Ataques oportunistas:** es un tipo de ataque a un sistema “software” que aprovecha algún punto vulnerable que no ha sido identificado previamente.

**“Bearer Token”:** es un “token” de acceso seguro que contiene las credenciales de seguridad para la gestión de la sesión e identificación del usuario.

**Bombas lógicas de tiempo:** es un ataque malicioso programado a una fecha precisa a los sistemas.

**Cámara de Compensación Automatizada (ACH):** es un tipo de tecnología que permite procesar transacciones entre bancos y otras instituciones financieras.

**Indicadores Claves de Rendimiento (KPI):** es una medida del nivel de rendimiento o desempeño de un proceso.

**Ingeniería social:** es una práctica ilegal que consiste en obtener información privada o confidencial a través de la manipulación de las personas y sus cuentas de usuario.

**WYSIWYG:** lo que ves es lo que tienes (“What You See Is What You Get”), esto se aplica a los procesadores de texto y otros editores de texto con formato que permiten escribir un documento mostrando directamente el resultado final.

Referencias bibliográficas

Dabir, A. and A. Matrawy. Bottleneck Analysis of Traffic Monitoring using Wireshark. in Innovations in Information Technology, 2007. IIT 07. 4th International Conference on. 2007.

Eber, D. [donEber]. (2021, junio 17). Instalación de Git en Windows paso a paso | [2021| 2022]. Youtube. <https://www.youtube.com/watch?v=cYLapo1FFmA>

Francesco Fusco, L.D., Joseph Gasparakis. Towards Monitoring Programmability in the Future Internet: challenges and solutions. 2010.

G. V. Ríos, John A. Bohada e I. A. Delgado, “Gestión de seguridad de la información en las organizaciones”, en Investigación e Innovación en Ingeniería del Software. Medellín: Tecnológico de Antioquia, vol. 2, 2018, pp. 111-121.

Git. Git--fast-version-control (2022). <https://git-scm.com/download/win>

M. Seyyar, F. Çatak and E. Gül, “Detection of attack-targeted scans from the Apache HTTP Server access logs”, Appl. Comput. Inf., vol. 14, N.º 1, pp. 28-36, 2018.

Microsoft. Microsoft Message Analyzer Operating Guide. [En línea] 21 de junio de 2022. [Citado el: 21 de junio de 2022.] <https://technet.microsoft.com/enus/library/jj649776.aspx>

NodeJs. [En línea] 17 de mayo de 2022. [Citado el: 17 de mayo de 2022.] <https://nodejs.org/es/download/>

OWASP Zed Attack Proxy (ZAP). [En línea] 15 de mayo de 2022. [Citado el: 15 de mayo de 2022.] <https://www.zaproxy.org>

OWASP Juice Shop. [En línea] 15 de mayo de 2022. [Citado el: 15 de mayo de 2022.] <https://owasp.org/www-project-juice-shop/>

OWASP. 2021. [En línea] 11 de mayo de 2022. [Citado el: 11 de mayo de 2022.] <https://owasp.org/>

OWASP. Top Ten. [En línea] 10 de mayo de 2022. [Citado el: 10 de mayo de 2022.] <https://owasp.org/www-project-top-ten/>

Pereira, C. P. T.-D. [DavidPereira]. (2021, marzo 4). Curso de Pentesting a Aplicaciones WEB (Video 1) OWASP Juice Shop. Youtube. <https://www.youtube.com/watch?v=6eAJ1dinee0>

Pries, R., et al. Traffic Measurement and Analysis of a Broadband Wireless Internet Access. In Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th. 2009.

PRTG Network Monitor. [En línea] 21 de junio de 2022. [Citado el: 21 de junio de 2022.] <https://www.paessler.com/manuals/prtg>

R. Rodriguez, H. Alberto. “The Concept of Corporate Strategy. Innovar: Revista de Ciencias Administrativas y Sociales”. 19, 2009.

S. A. Pineda, John A. Bohada y M. L. Pineda, “Ingeniería Social en Instituciones de Educación Superior”, Revista Colombiana de Tecnologías de Avanzada, vol. 2, N.º 32, pp. 52-61, 2018.

Security Knowledge Framework. [En línea] 15 de mayo de 2022. [Citado el: 15 de mayo de 2022.] <https://www.securityknowledgeframework.org/>

Shaoqiang, W., X. DongSheng, and Y. ShiLiang. Analysis and application of Wireshark in TCP/IP protocol teaching. in E-Health Networking, Digital Ecosystems and Technologies (EDT), 2010 International Conference on. 2010.

UskoKruM [UskoKruM2010]. (2021, agosto 19). Cómo Descargar e Instalar Node.js en Windows 10 | Explicación Sencilla . Youtube. <https://www.youtube.com/watch?v=ipmhBYqIP44>

V. Mouli and K. P. Jevitha, “Web Services Attacks and Security- ASystematic Literature Review”, Procedia Comp. Sci., vol. 93, pp. 870-877, 2016.

wstg: The Web Security Testing Guide is a comprehensive Open Source guide to testing the security of web applications and web services. (s/f).

Xinyu, X. and S. Mishra. Where is the tight link in a home wireless broadband environment? in Modeling, Analysis & Simulation of Computer and Telecommunication Systems, 2009. MASCOTS 09. IEEE International Symposium on. 2009.

Créditos

| Nombre | Cargo | Centro de Formación y Regional |
| --- | --- | --- |
| Claudia Patricia Aristizábal | Responsable del Ecosistema | Dirección General |
| Rafael Neftalí Lizcano Reyes | Responsable de Línea de Producción | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Danny Alejandro Solano | Experto temático | Centro de teleinformática y Producción Industrial - Regional Cauca |
| Paula Andrea Taborda Ortiz | Diseñadora instruccional | Centro de la Industria, la Empresa y Los Servicios CIES - Regional Norte de Santander |
| Carolina Coca Salazar | Asesora metodológica | Centro de Diseño y Metrología - Regional Distrito Capital |
| Sandra Patricia Hoyos Sepúlveda | Corrección de estilo | Centro de Diseño y Metrología - Regional Distrito Capital |
| Antonio Vecino Valero | Diseño web | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Francisco José Lizcano Reyes | Desarrollador Fullstack | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Wilson Andrés Arenales Cáceres | Animador y Producción audiovisual | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Emilsen Alfonso Bautista | Actividad Didáctica | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Zuleidy María Ruiz Torres | Validador de Recursos Educativos Digitales | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Luis Gabriel Urueta Álvarez | Validador de Recursos Educativos Digitales | Centro Industrial del Diseño y la Manufactura - Regional Santander |
| Daniel Ricardo Mutis Gómez | Evaluador para contenidos inclusivos y accesibles | Centro Industrial del Diseño y la Manufactura - Regional Santander |