**FORMATO PARA EL DESARROLLO DE COMPONENTE FORMATIVO**

| **PROGRAMA DE FORMACIÓN** | Aseguramiento de la calidad de software |
| --- | --- |

| **COMPETENCIA** |  | **RESULTADOS DE APRENDIZAJE** |  |
| --- | --- | --- | --- |

| **NÚMERO DEL COMPONENTE FORMATIVO** | 20 |
| --- | --- |
| **NOMBRE DEL COMPONENTE FORMATIVO** | Vulnerabilidades de seguridad |
| **BREVE DESCRIPCIÓN** | Un aspecto importante en el desarrollo de software, es el aseguramiento de la seguridad en cada una de las etapas del proceso de desarrollo incluyendo los despliegues en cada ambiente: desarrollo, pruebas y producción, logrando calidad y confiabilidad de las aplicaciones desarrolladas. |
| **PALABRAS CLAVE** | Seguridad, OWASP, configuración, diagnóstico y ciberseguridad |

| **ÁREA OCUPACIONAL** | 2 - Ciencias naturales, aplicadas y relacionadas |
| --- | --- |
| **IDIOMA** | Español |

1. **TABLA DE CONTENIDO**

Introducción

1. Marco de referencia en gestión de la seguridad

1.1 Seguridad en el análisis de requerimientos

1.2 Seguridad en el proceso de diseño.

1.3 Seguridad en el proceso de codificación.

1.4 Seguridad en el proceso de pruebas

1.5 Seguridad en el proceso de despliegue y mantenimiento

1. Matriz de control de acceso.
2. Seguridad perimetral.
3. Protocolos de comunicación segura.
4. Pruebas de vulnerabilidad
5. **INTRODUCCIÓN**

Bienvenidos al componente formativo: vulnerabilidades de seguridad.

Cuando se habla de desarrollo de *software*, se deben tener en cuenta varios aspectos para no solo desarrollar pensando en solucionar una necesidad de un cliente, sino también en hacerlo contemplando características que otorgan calidad a los productos desarrollados. La evolución de la tecnología, los medios de comunicación y el acceso a la información han hecho que la seguridad de las aplicaciones se convierta en un aspecto de gran importancia, otorgando confiabilidad a usuarios y clientes al resguardar la información. Por lo tanto, este componente se enfocará en cómo evaluar y diagnosticar este importante aspecto en el desarrollo de *software.*

La gestión efectiva de la seguridad en el desarrollo de *software* comprende una serie de etapas que van desde el análisis de requerimientos hasta el despliegue y mantenimiento, asegurando la integración de prácticas de seguridad a lo largo de todo el proceso. Además, herramientas como la matriz de control de acceso, estrategias de seguridad perimetral, y el uso de protocolos de comunicación segura son esenciales para fortalecer la defensa contra amenazas externas. Las pruebas de vulnerabilidad desempeñan un papel clave en la identificación y mitigación de posibles riesgos, garantizando así que las aplicaciones no solo cumplan con las expectativas funcionales, sino que también mantengan la integridad y confidencialidad de la información frente a potenciales vulnerabilidades. Este enfoque integral hacia la seguridad es fundamental para desarrollar software que sea robusto, confiable y seguro para los usuarios finales.

1. **DESARROLLO DE CONTENIDOS:**
2. **Marco de Referencia en Gestión de la seguridad**

El *software* y la forma en que se accede a las aplicaciones exigen que arquitectos, diseñadores y programadores se centren en la seguridad de estas. Por lo tanto, es importante tener en cuenta que la seguridad no solo implica la configuración de acceso, las redes o los usuarios, sino que debe considerarse durante todo el proceso de desarrollo de las aplicaciones. Esto incluye seguir estándares de programación y adoptar buenas prácticas de desarrollo, como la norma ISO/IEC 27034:2011, COBIT y OWASP.

**Figura 1**

Ciclo de vida de desarrollo de *software*







Dentro de este marco de referencia se quiere enfocar en algunos aspectos relevantes del aseguramiento desde el desarrollo del *software.*

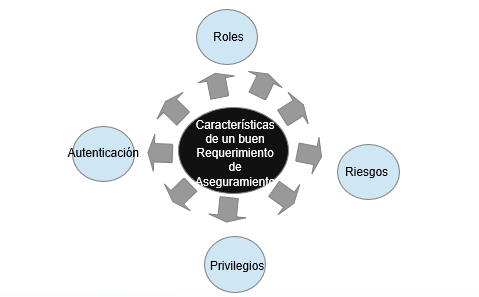
1. **Seguridad en el análisis de requerimientos**



Con respecto a la seguridad en esta etapa del desarrollo, es crucial tener en cuenta las normas y políticas. De esta manera, los desarrolladores sabrán desde el principio qué funcionarios o roles tienen permitido realizar ciertas acciones y cuáles no. Esto facilitará la comprensión de la gestión de usuarios con sus roles y autenticación en las aplicaciones. Se deben evaluar posibles riesgos tales como la comunicación de datos con terceros, políticas de confidencialidad, seguridad de los datos de usuarios que acceden y se registran en la aplicación, y los respaldos de seguridad de la información, entre otros. Además, será posible analizar la estructura para auditar cambios en la información, identificando quién y cuándo se agregaron o modificaron datos. Los requisitos generales a considerar incluyen: autenticación, asignación de roles, aprobación de privilegios y evaluación de riesgos.

**Figura 2**

*Requerimientos de Aseguramiento de calidad*



1. **Seguridad en el proceso de diseño**

En la etapa de diseño, es fundamental considerar los procesos y las definiciones de los requisitos de seguridad previamente identificados durante el proceso. Esto incluye, por ejemplo, establecer la periodicidad de las copias de seguridad, el tipo de contraseñas a utilizar (ya sean largas o cortas) y todas sus características, las cuales deberán actualizarse en periodos de tiempo más cortos, además de establecer el cifrado de las comunicaciones y de los datos.

Durante esta etapa, también se deben tener en cuenta las posibles vulnerabilidades que pueden surgir dependiendo de la arquitectura y el diseño de la aplicación. Por ejemplo:



Por lo tanto, es esencial diseñar un modelo de seguridad o de amenazas. Este modelo tiene como objetivo organizar, capturar y analizar los riesgos y vulnerabilidades identificados para tomar decisiones de manera rápida y ejercer un control adecuado implementando contramedidas de forma efectiva en el proceso.

Para elaborar un modelo de amenazas, es importante tener en cuenta los siguientes pasos:



En resumen, en el proceso de diseño, un modelo de seguridad es capaz de identificar las siguientes vulnerabilidades y posibles amenazas, como se muestra a continuación:



En esta etapa, también se identifican requisitos similares a los de la primera fase, que incluyen: acceso a los componentes, administración del sistema, auditoría, gestión de sesiones, registros históricos, manejo de errores y excepciones, y separación de funciones.

1. **Seguridad en el proceso de codificación**

En el proceso de desarrollo de *software*, es crucial considerar cómo los desarrolladores modifican y construyen los distintos componentes de las aplicaciones. Por lo tanto, resulta fundamental estandarizar las prácticas de codificación, documentación e identificación de ciertos requisitos, que incluyen:



En esta etapa, el objetivo es facilitar el trabajo en equipo, evitando reprocesos y pérdida de trabajo. Por ello, es crucial utilizar herramientas que permitan sincronizar el trabajo y guardar la información de forma colaborativa. Además, se recomienda emplear buenas prácticas estandarizadas por organizaciones como OWASP o CERT, las cuales se detallarán más adelante. No obstante, es importante considerar las buenas prácticas para una adecuada codificación, tales como:



1. **Seguridad en el proceso de pruebas**

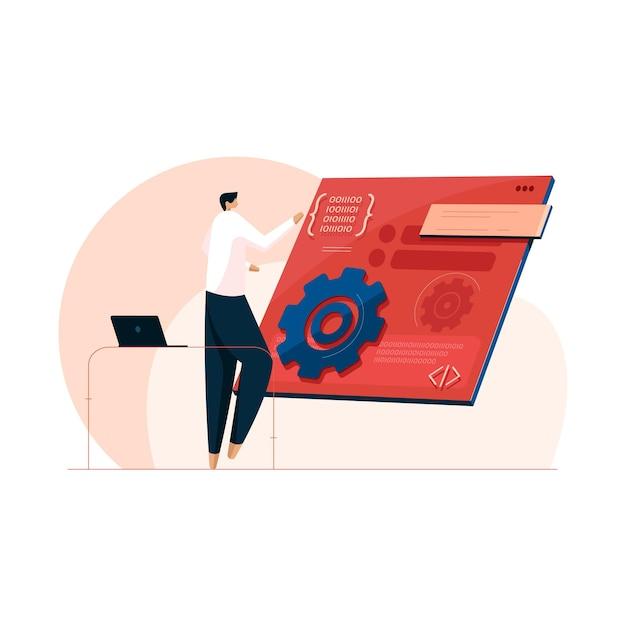
En la etapa de pruebas, se pueden analizar las vulnerabilidades siguiendo el modelo de amenazas diseñado en la etapa anterior. Las pruebas se centrarán en cada uno de los aspectos del modelo, considerando la configuración de los entornos, la codificación realizada por los desarrolladores y la arquitectura de la aplicación. De esta manera, será posible identificar y corregir vulnerabilidades a tiempo. Una de las técnicas utilizadas para estas pruebas es el *Fuzzing*, que consiste en enviar datos secuenciales o aleatorios a un *software* o aplicación con el objetivo de detectar vulnerabilidades no previstas. Además, es fundamental considerar el aseguramiento de la calidad, por lo cual se identifican los siguientes requerimientos:



1. **Seguridad en el proceso de despliegue y mantenimiento**

Una buena práctica consiste en asegurar que los servidores destinados a los despliegues de mantenimiento, desarrollo y producción mantengan configuraciones casi idénticas. La razón por la cual se dice "casi idénticas" es que existen ciertas configuraciones que pueden variar ligeramente debido al acceso a las aplicaciones, pero, en cuanto a características generales, deberían ser iguales. Esto permite que, al realizar pruebas, las aplicaciones se comporten de manera similar en diferentes entornos.





Además, es necesario configurar la red y gestionar el acceso de manera que solo las personas autorizadas puedan acceder a ella. Es importante tener en cuenta la arquitectura diseñada en etapas anteriores para entender bien los tipos de configuración. Configurar los accesos y la red para despliegues directos en la nube o en un servidor local implica consideraciones diferentes. OWASP también ofrece guías sobre cómo probar estos tipos de aseguramientos.

**Entidades de estandarización**

Existen entidades responsables de elaborar normativas de nivel técnico, que, con el apoyo de equipos técnicos compuestos por expertos, definen estándares. Estos estándares son técnicas establecidas en documentos publicados que facilitan la creación de un marco de trabajo para abordar problemas comunes en las organizaciones.

A continuación, se listan las principales entidades:



En la siguiente tabla, se proporciona una perspectiva global de algunas normas que abordan la definición de ciberseguridad.

**Tabla 1**

*Normas relacionadas con la ciberseguridad*

| **N°** | **Origen** | **Documento** | **Término** | **Organización** | **CDI** | **Intensión** | **Motivación** | **Amenazas** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **01** | ISO/IEC  JTC1/SC27 | 27032 | Ciberseguridad | ISO | SÍ | Solo los activos destinados a internet | No diferenciación Entre actividades maliciosas o no | Sólo los activos virtuales conectados a internet, sin activos físicos |
| **02** | ISO/IEC  JTC1/SC27 | 27000 | Seguridad de la información | ISO | SÍ | Cualquier Origen de Riesgo en el Espacio cibernético | No diferenciación entre malicioso o no intencional | Cualquier Activo |
| **03** | UIT-T | X.1205 | Ciberseguridad | UIT | NO | Cualquier Origen de Riesgo en el Espacio cibernético | No diferenciación entre actividades maliciosas o no | Cualquier Activo |
| **04** | NIST | SP 800-39 | Ciberseguridad | NIST | SÍ | Riesgo Originado exclusivamente en el espacio cibernético | Sólo cubre orígenes maliciosos (ataques cibernéticos | Sólo los activos virtuales conectados a internet, sin activos físicos |
| **05** | ISACA | Cybersecurity fundamentals | Ciberseguridad | ISACA | SI | Certificación CSX | Origen Malicioso | Activos en el Ciberespacio |

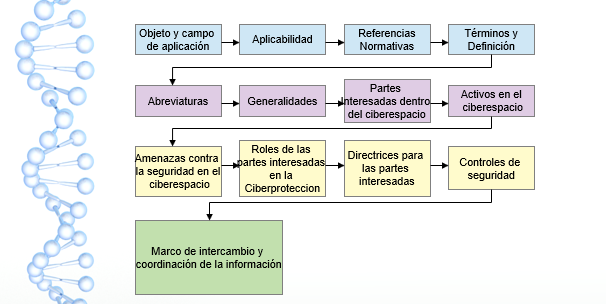
*Nota.* Gestión de la ciberseguridad según el ISO/IEC 27032:2012. ,2021)

La norma ISO/IEC 27032:2020 "Tecnología de la información - Técnicas de seguridad - Directrices para la ciberseguridad" establece un marco de trabajo diseñado para mejorar el estado de la ciberseguridad. Propone varios puntos estratégicos y técnicos necesarios para esta actividad, así como sus interdependencias con otros dominios de seguridad, incluyendo:



**Figura 3**

*Estructura de la norma ISO/IEC 27032:2020*



La norma se enfoca en la seguridad del ciberespacio o cuestiones de ciberseguridad, con un énfasis particular en establecer conexiones entre diferentes brechas de la web. Este documento ofrece una guía técnica para abordar los riesgos comunes de ciberseguridad.

Dado el aumento en la frecuencia y la sofisticación de los ciberataques, se reconoce claramente la necesidad de proteger los recursos y activos. Surge entonces la importante pregunta:



Aunque es esencial elegir un marco de referencia y trabajar con él, es importante recordar que son solo una guía y no ofrecen soluciones definitivas. El enfoque en el análisis de riesgos es también crucial; cada marco propone una serie de controles entre los cuales se debe elegir los más aplicables al entorno específico y, si es necesario, realizar adopciones o ajustes de otros marcos de referencia.

Los marcos de referencia más utilizados y desarrollados incluyen propósitos particulares para su adopción y una breve descripción, aunque no se han detallado específicamente aquí.

Los marcos de referencia más utilizados son:



1. **Matriz de control de acceso**

La matriz de control es un documento diseñado para organizar los roles y permisos analizados desde la etapa de desarrollo, específicamente en la fase de requerimientos. Constituye una manera estructurada de definir quién va a realizar qué acciones dentro de la aplicación. Esta herramienta es particularmente útil para empresas grandes que gestionan diversas aplicaciones para llevar a cabo sus operaciones diarias.

| Persona trabajando html en computadora | Uno de los inconvenientes al desarrollar *software* es realizar pruebas sin considerar adecuadamente el acceso o permisos que cada usuario debe tener, dejando la organización de estos aspectos para la etapa final. Este enfoque puede resultar en reprocesos y en una evaluación insuficiente de la seguridad de la aplicación con respecto al usuario interno. Es crucial reconocer que el usuario interno de la organización representa el primer punto de vulnerabilidad; por tanto, la seguridad interna debe ser considerada un elemento importante dentro del análisis y diseño de las aplicaciones. |
| --- | --- |

A continuación, se muestra un ejemplo de una matriz reducida para ilustrar el enfoque que se debe adoptar y los aspectos que se deben considerar en su implementación:

**Tabla 2**

Matriz de control de acceso

| **Rol** | **Gestión de usuarios** | **Crear cuenta** | **Actualizar cuenta** | **Eliminar cuenta** | **Crear registro** | **Consultar estado de cuenta** |
| --- | --- | --- | --- | --- | --- | --- |
| **Administrador** |  |  |  |  |  |  |
| **Auxiliar contable** |  |  |  |  |  |  |
| **Operador** |  |  |  |  |  |  |

Esta matriz pertenece a una aplicación contable sencilla y muestra los roles de forma vertical en el lado izquierdo, mientras que las funcionalidades se disponen horizontalmente en la parte superior. Ahora procederemos a gestionarla, especificando quién podrá realizar qué acción o quién tendrá acceso a cada una de las funcionalidades. De esta manera, podemos ejercer control sobre la gestión de roles y permisos, determinando el nivel de acceso que cada usuario tiene en la aplicación.

**Tabla 3**

Gestión de Matriz de control de acceso

| **Rol** | **Gestión de usuarios** | **Crear cuenta** | **Actualizar cuenta** | **Eliminar cuenta** | **Crear registro** | **Consultar estado de cuenta** | **Registro de pagos** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Administrador** | x | x | x | x | x | x | x |
| **Auxiliar contable** |  |  |  |  | x | x | x |
| **Operador** |  |  |  |  |  |  | x |

En la gestión de la matriz, se observa que el administrador podrá realizar todas las funciones, mientras que el auxiliar contable solo podrá ejecutar tareas como crear registros y consultar el estado de las cuentas. Por otro lado, el operador solo tendrá la capacidad de registrar pagos. Es recomendable revisar los accesos mediante la matriz de acceso de forma periódica para garantizar la seguridad y la correcta asignación de permisos.

1. **Seguridad perimetral**

En el ámbito de la seguridad informática, los sistemas de seguridad perimetral buscan prevenir el acceso no autorizado a la red, así como a los sistemas e información que contiene.



Para profundizar en el tema, se invita a revisar el siguiente video:



1. **Protocolos de comunicación segura**

Cuando navegamos en internet a través de nuestro navegador preferido, se produce un intercambio de información con las distintas páginas web que visitamos. En muchas ocasiones, se comparte información que el usuario no percibe, mientras que otras veces se trata de información que se comparte de manera consciente, como cuando un usuario se registra o se suscribe.



Existen diversos tipos de protocolos diseñados para esta protección, los cuales se mencionan a continuación:



1. **Pruebas de vulnerabilidad**

Se trata de pruebas de *software* realizadas para evaluar los riesgos de seguridad en un *software*, con el objetivo de minimizar las posibles amenazas y prevenir futuros ataques cibernéticos que puedan explotar vulnerabilidades en cada sistema instalado.

A continuación, se presentará una metodología práctica para identificar vulnerabilidades. Aunque no se sepa cómo explotarlas, es importante ser conscientes de que existen individuos que podrían hacerlo, perjudicando así a nosotros, a nuestros clientes y a sus reputaciones. Por razones de seguridad respecto a las empresas evaluadas, se utilizará como ejemplo el sitio web de una entidad estatal, cuyo nombre y dirección web se mantendrán en reserva. La imagen será editada para no comprometer la reputación de la entidad.

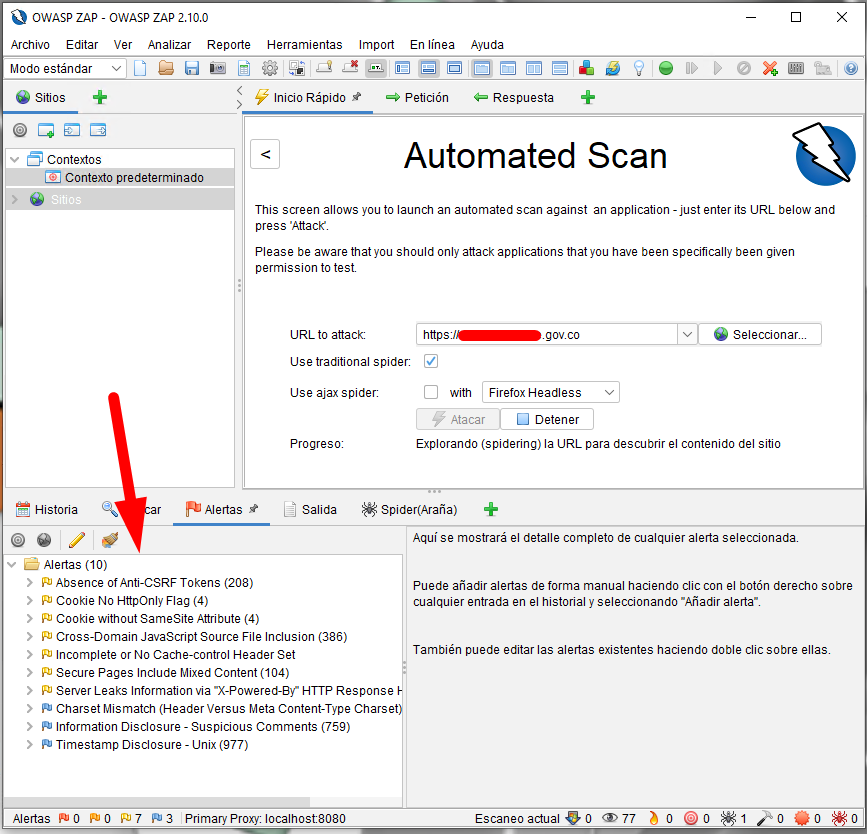
| Document with solid fill | **Instalación OWASP ZAP**  Para llevar a cabo este proceso, es necesario instalar la aplicación OWASP ZAP en un sistema operativo Windows 10. Para ello, puede descargar el instructivo que le compartimos para este proceso. |
| --- | --- |

Esta herramienta está diseñada para realizar pruebas de vulnerabilidad en aplicaciones y requiere tener instalado el JDK de JAVA como prerrequisito. Una vez instalado el JDK puede continuar descargando el instalador para el sistema operativo indicado.

A continuación, se presenta un ejemplo de prueba.

**Figura 4.**

Ejemplo de prueba



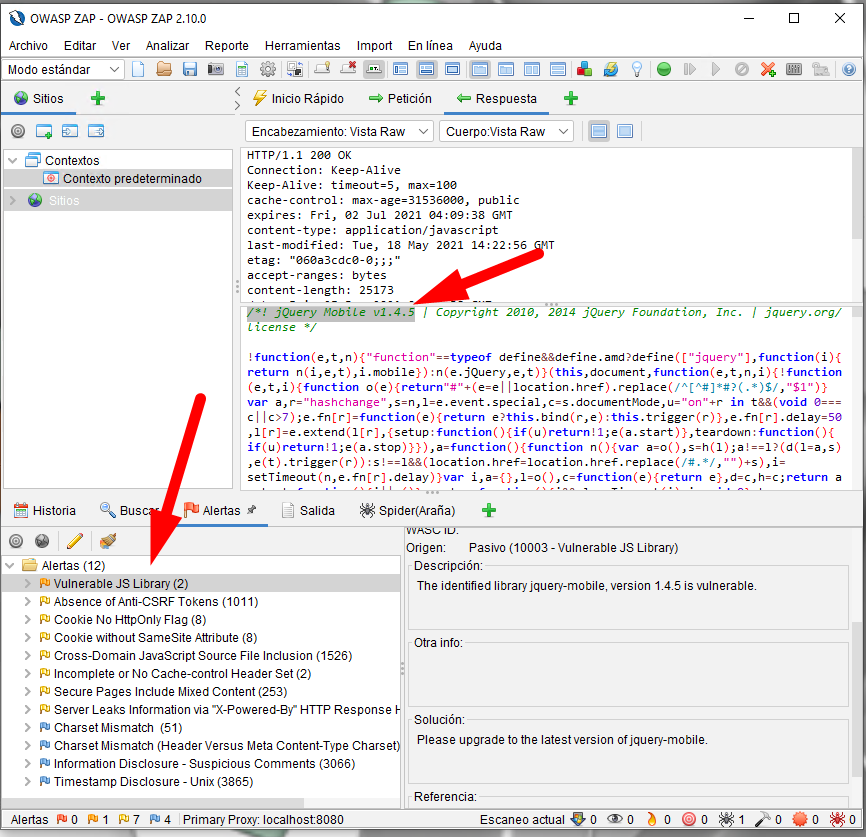
A partir de la prueba presentada, se explicará un poco las vulnerabilidades encontradas:



Examinemos un caso crítico que también es fácilmente explotable.

**Figura 5**

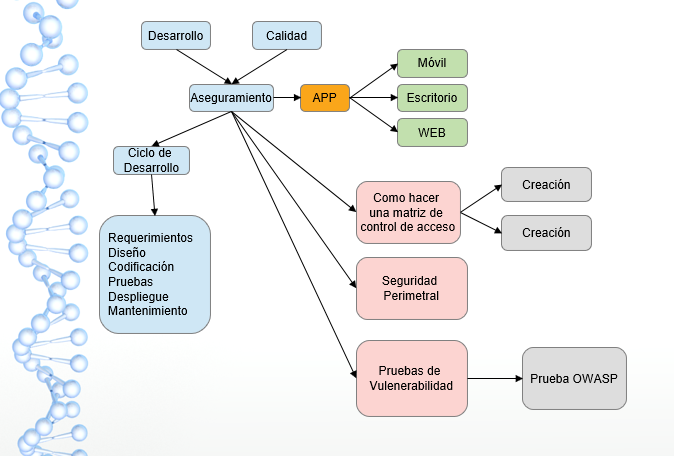
*OWASP ZAP 2.10.0*



El sitio web está utilizando la librería jQuery Mobile versión 1.4.5, la cual ya ha sido identificada por la comunidad jQuery como vulnerable. Es probable que en los foros de dicha comunidad se discuta cómo se detectó y esto podría indicar cómo vulnerar este sistema. Sin embargo, la solución propuesta en la imagen es simplemente actualizar la librería a una versión más reciente y segura.

1. **SÍNTESIS**

A continuación, se presenta una síntesis de la temática estudiada en el componente formativo.



1. **ACTIVIDADES DIDÁCTICAS (OPCIONALES SI SON SUGERIDAS)**

| **DESCRIPCIÓN DE ACTIVIDAD DIDÁCTICA** | |
| --- | --- |
| **Nombre de la actividad** | Vulnerabilidades de seguridad |
| **Objetivo de la actividad** | Fortalecer la comprensión de las características fundamentales de conceptos seleccionados en seguridad y aseguramiento de software. |
| **Tipo de actividad sugerida** | Arrastrar y soltar el concepto correcto con la descripción correspondiente relacionados con el aseguramiento de software. |
| **Archivo de la actividad (Anexo donde se describe la actividad propuesta)** | Anexo documento en Word llamado Actividad didáctica 1 |

1. **MATERIAL COMPLEMENTARIO**

| **Tema** | **Referencia APA del Material** | **Tipo de material**  **(Video, capítulo de libro, artículo, otro)** | **Enlace del Recurso o**  **Archivo del documento o material** |
| --- | --- | --- | --- |
| **3.Seguridad perimetral.** | ComputerHoy.com. (2015). ¿Qué es un firewall o Cortafuegos?  <https://www.youtube.com/embed/3q2ENiVBAy8> | Video de YouTube | <https://www.youtube.com/embed/3q2ENiVBAy8> |
| **3.Seguridad perimetral.** | VpnMentor. (2021, Enero 12). VPN Guide for Newbies. Blogpost. https://es.vpnmentor.com/blog/que-es-una-vpn-guia-sobre-vpns-para-principiantes/ | Blog | https://es.vpnmentor.com/blog/que-es-una-vpn-guia-sobre-vpns-para-principiantes/ |
| **3.Seguridad perimetral.** | López, A.(2021). ¿Qué es un Honeypot en ciberseguridad? ¿para qué sirve? Tipos de Honeypots.  <https://www.youtube.com/embed/SgH9rWB9ivQ> | Video de YouTube | <https://www.youtube.com/embed/SgH9rWB9ivQ> |
| 4. **Protocolos de comunicación segura.** | Azuax.C.(2017). Conceptos Fundamentales del protocolo HTTP y HTTPS. <https://www.youtube.com/embed/ARmQMSeU9fU> | Video de YouTube | <https://www.youtube.com/embed/ARmQMSeU9fU> |
| **5.Pruebas de vulnerabilidad** | Roel.A(2021).  Como descargar e instalar Java JDK16 en Windows 10-2021. <https://www.youtube.com/watch?v=hCBEavs08as> | Video de YouTube | <https://www.youtube.com/watch?v=hCBEavs08as> |

1. **GLOSARIO**

| **TÉRMINO** | **SIGNIFICADO** |
| --- | --- |
| Autenticación: | es el proceso que debe seguir un usuario para tener acceso a los recursos de un sistema o de una red de computadores. Este proceso implica identificación (decirle al sistema quién es) y **autenticación** (demostrar que el usuario es quien dice ser). |
| Sitio Web: | es un conjunto de páginas **web** accesibles a través de **internet**, convenientemente enlazadas y con una finalidad concreta. |
| Sintaxis: | según su **definición** la **sintaxis** es la “Parte de la gramática que estudia el modo en que se combinan las palabras y los grupos que estas forman para expresar significados, así como las relaciones que se establecen entre todas esas unidades” |
| Delegar: | dar [una persona o un organismo] un poder, una función o una responsabilidad a alguien para que los ejerza en su lugar o para obrar en representación suya. |
| Web: | conjunto de información que se encuentra en una dirección determinada de internet |
| Interfaz: | como **interfaz** designamos, en informática, la conexión física y funcional que se establece entre dos aparatos, dispositivos o sistemas que funcionan independientemente uno del otro. En este sentido, la comunicación entre un ser humano y una computadora se realiza por medio de una **interfaz** |
| API: | una **API** es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el *software* de las aplicaciones. **API** significa interfaz de programación de aplicaciones. Las **API** permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados. |
| Aplicación: | es un programa **informático** diseñado como una herramienta para realizar operaciones o funciones específicas. Generalmente, son diseñadas para facilitar ciertas tareas complejas y hacer más sencilla la experiencia **informática** de las personas |
| *Browser:* | es el término inglés que se utiliza para identificar a un navegador web o navegador de Internet. Consiste en un *software*, programa o incluso aplicación, que ofrece al usuario el acceso a la Red |
| Servidor: | es un conjunto de computadoras capaz de atender las peticiones de un cliente y devolverle una respuesta en concordancia. |

1. **REFERENCIAS BIBLIOGRÁFICAS:**

Atico34 (2021). Seguridad Perimetral informática., <https://protecciondatos-lopd.com/empresas/seguridad-perimetral-informatica/#Definicion_de_seguridad_perimetral>

Calder, A. (2018). NIST Cybersecurity Framework: Una guía de bolsillo . IT Governance Publishing Ltd.

Cano, J. (2011). Ciberseguridad y ciberdefensa: dos tendencias emergentes en un contexto global. SISTEMAS (ASOCIACION COLOMBIANA DE INGENIEROS DE SISTEMAS), 119, 4-7.

Firma-e. (2021). ¿Qué es un SGSI – Sistema de Gestión de Seguridad de la Información?. <https://www.firma-e.com/blog/que-es-un-sgsi-sistema-de-gestion-de-seguridad-de-la-informacion>

Hackmetrix (2021). Matriz de control de accesos: Qué es y cómo hacerla paso a paso. ,<https://blog.hackmetrix.com/matriz-de-accesos/>

ISO/IEC. (2020). Tecnologías de la información. Técnicas de seguridad. directrices para ciberseguridad (Num. 27032) <https://tienda.icontec.org/gp-tecnologias-de-la-informacion-tecnicas-de-seguridad-directrices-para-ciberseguridad-gtc-iso-iec27032-2020.html>

Soriano, M. (2014). Seguridad en redes y seguridad de la información. Obtenido de <http://improvet.cvut.cz/project/download/C2ES/Seguridad_de_Red_e_Informacion.pdf>.

1. **CONTROL DEL DOCUMENTO**

|  | **Nombre** | **Cargo** | **Dependencia**  ***(Para el SENA indicar Regional y Centro de Formación)*** | **Fecha** |
| --- | --- | --- | --- | --- |
| Autor (es) | David Eduardo Lozada Cerón | Experto Temático | Regional Cauca, Centro de Tele informática y Producción Industrial | Diciembre de 2021 |
| Peter Pinchao | Experto temático | Regional Cauca- Centro Teleinformática y Producción Industrial | Diciembre de 2021 |

1. **CONTROL DE CAMBIOS**

|  | Nombre | Cargo | Dependencia | Fecha | Razón del Cambio |
| --- | --- | --- | --- | --- | --- |
| Autor (es) | Olga Constanza Bermúdez Jaimes | Responsable Línea de Producción Antioquia | Centro de Servicios de Salud | Marzo 2024 | Actualización |
|  | Paola Alexandra Moya | Evaluadora instruccional | Centro de Servicios de Salud | Marzo 2024 | Actualización |