ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

Desarrollo de un módulo de seguridad basado en OSSTMM y OWASP para mitigar y controlar la seguridad en aplicaciones web caso de estudio: Sistema Médico Escuela Superior Politécnica De Chimborazo

HERNÁN DARÍO CENTENO AULLA

Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo, presentado ante el Instituto de Posgrado y Educación Continua de la ESPOCH, como requisito parcial para la obtención del grado de:

MAGÍSTER EN SEGURIDAD TELEMÁTICA

RIOBAMBA – ECUADOR

Enero 2024

DECLARACIÓN DE AUTENTICIDAD Y CESIÓN DE DERECHOS DE AUTOR

Yo, Hernán Darío Centeno Aulla, declaro que el presente **Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo**, es de mi autoría y que los resultados del mismo son auténticos y originales. Los textos constantes en el documento que provienen de otra fuente están debidamente citados y referenciados.

Como autor, asumo la responsabilidad legal y académica de los contenidos de este proyecto de investigación de maestría, el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

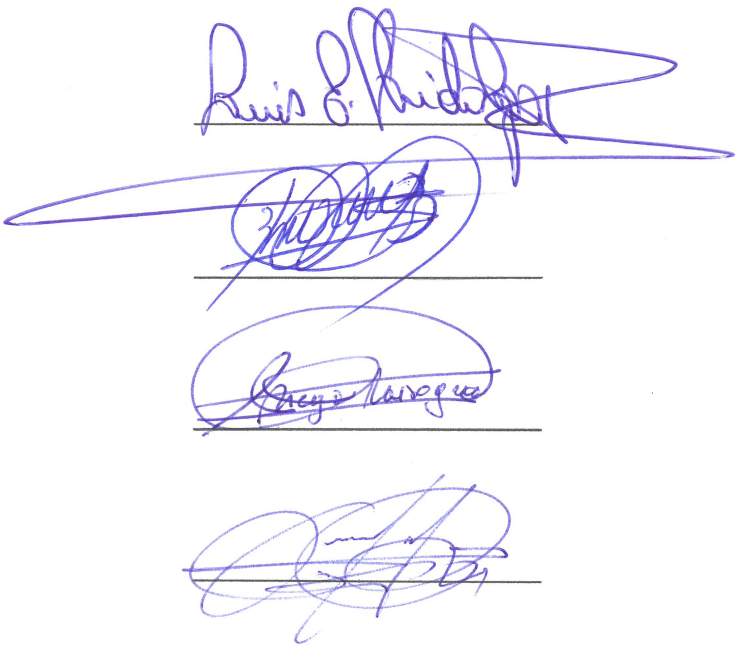
Hernán Darío Centeno Aulla

CI: 060412055-0

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

EL TRIBUNAL DE TRABAJO DE TITULACIÓN CERTIFICA QUE:

El **Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo**, titulado Desarrollo de un módulo de seguridad basado en OSSTMM y OWASP para mitigar y controlar la seguridad en aplicaciones web caso de estudio: Sistema Médico Escuela Superior Politécnica De Chimborazo, de responsabilidad del señor Hernán Darío Centeno Aulla ha sido minuciosamente revisado por los Miembros del Tribunal del trabajo de titulación, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal autoriza su presentación.



Ing. Luis Eduardo Hidalgo Almeida; Ph.D.
PRESIDENTE

Ing. Byron Ernesto Vaca Barahona, Ph.D.
DIRECTOR

Ing. Diego Francisco Caisaguano Villa, Mgtr.
MIEMBRO

Ing. Marco Vinicio Ramos Valencia, Mgtr.
MIEMBRO

Enero - 2024

DEDICATORIA

Dedico este presente trabajo de manera especial a mis hijos Jean Sebastián y Gael Nicolas que siempre serán el motivo para ser mejor persona y profesional, de igual manera mis padres Luis Centeno y Luz Aulla que fruto de su esfuerzo y ejemplo de superación me motivaron siempre a salir adelante.

Hernán

AGRADECIMIENTO

En primer lugar, quiero agradecer a la Escuela Superior Politécnica de Chimborazo por impartir el conocimiento apropiado para ser un buen profesional y en segundo lugar quiero agradecer a mis padres por su apoyo incondicional.

Hernán

TABLA DE CONTENIDO

CAPÍTULO IV

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

ÍNDICE DE GRÁFICOS

RESUMEN

En el presente trabajo se expone la necesidad de enfrentar las amenazas o vulnerabilidades que presentan los sistemas para la web, estos ataques tienen como objetivo la sustracción de información que albergan las bases de datos de dichos sistemas, razón por la cual se debe implementar métricas de seguridad que garanticen la seguridad de la información que es el órgano vital para cualquier entidad, motivo por el cual se desarrolló un módulo de seguridad para sistemas en la web el mismo que fue aplicado en el sistema médico de la ESPOCH, este módulo tiene como finalidad garantizar la integridad de los datos que albergan dichos sistemas en sus bases de datos, para el desarrollo del módulo propuesto se basó en las metodologías OWASP y OSSTM. El módulo de seguridad está constituido por cuatro componentes los mismos que a su vez se clasifican en dos grupos según la funcionalidad que cumple cada uno de esta manera se tiene los componentes que interactúan directo con el sistema y los componentes que son aplicados a los servidores de producción. Para el desarrollo de los componentes del primer módulo se hizo del Framework de Angular, estos componentes actúan como una capa intermedia entre la vista de usuario y la lógica de negocios del sistema médico. Para el segundo grupo de componentes se hizo la implementación de certificados SSL y un script en el servidor de base de datos que permite la creación de copias de seguridad de la base de datos del sistema médico. Para la evaluación del módulo de seguridad implementado se levantaron cuatro ambientes de pruebas es decir uno por cada componente del módulo desarrollado, que por medio del método estadístico de T-Student se hizo el análisis de los datos obtenidos, para este caso de estudio se trabajó con un nivel de confianza del 95%, con un nivel de significancia igual a 0.05, como resultado del estadístico T se obtiene un valor de P(T<=t) dos colas igual a 6.88224E-06 que en comparación al nivel de significancia se aprecia una diferencia significativa con los que se comprueba que la implementación del módulo desarrollado permite mejorar la seguridad en el sistema de salud de la ESPOCH.

Palabras Claves: <MÓDULO DE SEGURIDAD>, <INTEGRIDAD>, <FRAMEWORK>, <INTERFAZ DE USUARIO>, <CERTIFICADOS SSL>, <LÓGICA DE NEGOCIOS>, <SCRIPT>.

SUMMARY

In the current research the need to face the threats or vulnerabilities that present the systems for the web is exposed, these attacks are aimed at the theft of information that host the databases of these systems. For this reason it is necessary to implement security metrics to ensure the security of information that is the vital organ for any entity, therefore it was developed a security module for web systems which was applied in the medical system of the ESPOCH. This module aims to ensure the integrity of the data held by these systems in their databases, for the development of the proposed module was based on the OWASP and OSSTM methodologies. The security module is made up of four components, which in turn are classified into two groups according to the functionality that each one fulfills, thus we have the components that interact directly with the system and the components that are applied to the production servers. For the development of the components of the first module, the Angular Framework was used, these components act as an intermediate layer between the user view and the business logic of the medical system. For the second group of components, the implementation of SSL certificates and a script on the database server that allows the creation of backup copies of the medical system's database were carried out. For the evaluation of the implemented security module, four test environments were created, i.e. one for each component of the developed module, and the data obtained was analyzed using the T-Student statistical method. For this case study we worked with a confidence level of 95%, with a significance level equal to 0.05, as a result of the T statistic we obtained a value of P(T<=t) two-tailed equal to 6.88224E-06 which in comparison to the significance level shows a significant difference with which it is proved that the implementation of the developed module allows improving the security in the health system of ESPOCH.

Keywords: <SECURITY MODULE>, <INTEGRITY>, <FRAMEWORK>, <USER INTERFACE>, <SSL CERTIFICATES>, <BUSINESS LOGIC>, <SCRIPT>.

CAPÍTULO I

1. INTRODUCCIÓN

1.1. Planteamiento del problema

1.1.1. Situación problemática

En la actualidad existen varios métodos y técnicas que rompen con los esquemas de seguridad en los sistemas informáticos, por esta razón en la actualidad, las empresas tanto del sector público como del privado catalogan a la información como uno de los bienes más preciados (Monsalve et al., 2014).

De la misma forma existen empresas que no se preocupan por implementar medidas de seguridad informática o si lo hacen, solo consideran externalidades y no tienen en cuenta los riesgos que se puedan presentar al interior de las mismas (Vera, 2017). Esto provoca que la información que albergan los sistemas informáticos sea vulnerable (Voutssas, 2010) afirma que una vulnerabilidad es alguna característica o circunstancia de debilidad de un recurso informático la cual es susceptible de ser explotada por una amenaza, intencional o accidentalmente. Por su parte (Marco, 2013) afirma que si no hay amenazas no hay riesgo Si no hay vulnerabilidades no hay riesgo.

A lo largo del tiempo, el avance de los medios tecnológicos y de comunicación ha provocado el surgimiento de nuevos vectores de ataques y de nuevas modalidades delictivas (Mieres, 2009a), con la evolución de los sistemas para la web los delincuentes informáticos van desarrollado nuevas técnicas de ataque que permiten obtener información confidencial para las empresas u organizaciones, según (Guamán, 2011a) actualmente el 80% de los ataques informáticos son llevados a cabo por código malicioso y las configuraciones por defecto que se realizan en el desarrollo de un sistema Web hacen que el ataque sea una tarea sencilla.

Las vulnerabilidades de inyección son las más comunes en la capa aplicativa, los atacantes pueden aprovechar esta vulnerabilidad para extraer información de manera no autorizada o ejecutar código malicioso, ya que esta vulnerabilidad permite enviar sentencias SQL (SQL Injection), comandos para manipular la ejecución lógica de procesos en el sistema operativo, o evadir métodos de autenticación de manera no autorizada (López, 2015).

La introducción de estas sentencias o código malicioso se le conoce como inyecciones SQL, un ataque de inyección de SQL es un ataque que va dirigido a subvertir la intención original de la

solicitud mediante la presentación suministrada por el atacante sentencias SQL directamente en la base de datos (De La Quintana, 2013).

La introducción de código malicioso puede provenir desde la interfaz de usuario al hacer uso del sistema, por lo que cualquier usuario de un sistema informático web que tenga conocimientos en el manejo o administración de bases de datos puede introducir sentencias maliciosas que dejan expuesta toda la información que aloja la base datos.

Tomando en cuenta que el sistema de salud de la ESPOCH presenta varios perfiles de usuarios y en cada uno de dichos perfiles el usuario tiene varias opciones para realizar consultas por lo que se le presenta la opción de introducir en cajas de texto el criterio de búsqueda a realizar y obtener información específica según sea el caso. Bajo esta premisa el usuario puede introducir sentencias o código malicioso con la finalidad de dejar expuesta información adicional para la que está programado el sistema.

Otra consecuencia de tener campos abiertos para que el usuario ingrese caracteres a su criterio es que al introducir caracteres de diferentes tipos de datos al que espera el sistema, el mismo no pueda procesar la petición realizada por lo tanto el sistema tiende a entrar a un ciclo infinito botando como mensaje de error en la consola del navegador la ruta y puerto del servicio que no puede procesar, con esta información se puede realizar un análisis profundo a nivel de servicios web.

1.1.2. Formulación del problema

¿Cuáles con las vulnerabilidades que se puede mitigar con la integración de un módulo de seguridad en sistemas para la web?

1.1.3. Preguntas directrices o específicas de la investigación

¿Qué tipo de información es vulnerable en sistemas para la web?

¿Qué impacto tiene la alteración de información en sistemas para la web?

¿Qué métricas se debe tomar en cuenta para desarrollar un módulo de seguridad para sistemas en la web?

¿Cuáles son las ventajas al implementar un módulo de seguridad para sistemas en la web?

1.1.4. Justificación de la investigación

En la sociedad actual es incuestionable la relevancia de la Web, y existe una gran variedad de sitios web que brindan servicio a los usuarios (Perurena & Moráguez, 2013), en este sentido las organizaciones son completamente dependientes de la tecnología para llevar a cabo sus objetivos, las informaciones críticas son almacenadas, procesadas y transmitidas en formato digital (Montesino et al., 2013).

El primero de los tres principios de la seguridad de la información que aplicamos es la integridad, la cual nos permite garantizar que la información no ha sido alterada en su contenido, por tanto, es íntegra (Dussan, 2006).

Al mismo tiempo, son cada vez mayores las amenazas a la seguridad de la información que se transmite.

Es necesario entonces, crear diferentes mecanismos, dirigidos a garantizar la confidencialidad y autenticidad de la información (Marrero, 2003). Por esta razón nace la necesidad de crear métodos o estrategias que permita la seguridad de la información.

En la actualidad la mayoría de las empresas disponen de sistemas en la web en donde ofrecen sus servicios, siendo este el caso de la ESPOCH que gestiona y administra varios sistemas para la web los mismo que están disponibles a la comunidad politécnica, dichos sistemas de igual manera se encuentran expuestos a ataques informáticos.

Los ataques a los sistemas informáticos pueden provenir de varios factores ya sean estos internos o externos, estos ataques tienen como objetivo la sustracción de información que alberga los sistemas, por esta razón se debe implementar medidas de seguridad que detecte las vulnerabilidades que pueden presentar los sistemas para la web garantizando la integridad y confidencialidad de la información.

Para lograr con el objetivo planteado que es la detección y mitigación de vulnerabilidades en sistemas para la web, en este documento se plantea la elaboración de un módulo de seguridad el mismo que integrará medidas y normativas basadas en OSSTMM y OWASP sirviendo como base para mitigar falencias o brechas de seguridad que presentan los sitios web.

El módulo a ser desarrollado debe tener la capacidad de detectar las acciones que realiza el usuario final desde la vista del sistema y dar paso aquellas peticiones que no son mal intencionadas, en el

caso de ser acciones inapropiadas como la introducción de código malicioso el módulo de seguridad debe restringir el paso a estas peticiones.

En este sentido el módulo a ser desarrollo debe ser adaptable a cualquier sistema para la web y trabajar del lado del cliente por lo que se ha pensado hacer uso de AngularJS para la programación de dicho modulo, La estructura de AngularJS, obliga de manera ortodoxa a tener el código ordenado, ya que uso de controladores para enviar datos a la vista, modelos para base de datos, directivas, servicios o filtros, para la inyección de dependencias (Cangás, 2015).

1.1.5. Objetivo general de la investigación

Desarrollar de un módulo de seguridad basado en OSSTMM y OWASP que permita mitigar y controlar la seguridad en aplicaciones web caso de estudio: sistema médico de la ESPOCH.

1.1.6. Objetivos específicos de la investigación

- Realizar un estudio de las normativas y estrategias que establecen las metodologías OSSTMM y OWASP.
- Realizar un estudio de las principales vulnerabilidades que afectan a los sistemas para la web.
- Desarrollar el módulo de seguridad basado en métricas y normativas establecidas por las metodologías OSSTMM y OWASP que permita determinar las vulnerabilidades que presenta el sistema de salud de la ESPOCH.
- Determinar la eficiencia del sistema de seguridad desarrollado por medio de la evaluación aplicado al sistema médico de la ESPOCH.

1.1.7. Hipótesis

Con la implementación de un módulo de seguridad se mejora la integridad de la información del sistema de E-salud de la ESPOCH.

Dependiente: consecuencia Mejora la integridad de la información del sistema de E-salud de la ESPOCH.
Independiente: causa Implementación de un módulo de seguridad

CAPÍTULO II

2. MARCO TEÓRICO

2.1. Antecedentes del problema

En la actualidad la propagación del malware resulta mucho más rápida, sobre todo gracias al uso de internet. Además, ahora los virus no buscan en la mayoría de los casos la notoriedad, sino más bien todo lo contrario: permanecer ocultos en el sistema sin que la persona usuaria sepa que su ordenador está infectado (Zambrano & Valencia, 2017). De esta forma, las amenazas que se presentan en un sistema se convierten en la fuente o causa potencial de eventos o incidentes no deseados que pueden resultar en daño a los insumos informáticos de la organización y ulteriormente a ella misma (Tarazona & Cesar, 2007). Dando origen a nuevos personajes que utilizan los medios informáticos y el conocimiento sobre su funcionamiento como herramientas para delinquir y obtener algún beneficio.

Como consecuencia de todos estos cambios el volumen, naturaleza, disponibilidad y sensibilidad de la información que se intercambia a través de esta infraestructura se ha modificado y ha aumentado de manera muy significativa el desarrollo de sistemas de seguridad (Voutssas, 2010). Pues el uso de internet conlleva a que se desarrollen proyectos de seguridad informática que garanticen la integridad, disponibilidad y accesibilidad de la información (Zambrano & Valencia, 2017). Una manera de evitar los ataques informáticos es la prevención utilizando técnicas y herramientas para detectar vulnerabilidades potenciales que pueden ser aprovechadas por los atacantes.

El caso de estudio para aplicar un módulo de seguridad es el CASI de la ESPOCH, el cual cuenta con un sistema informático para la web, el cual contribuye a la gestión de las actividades realizadas dentro del Centro de Atención de Salud Integral. Dicho sistema, utiliza información digital la cual esta almacenada en una base de datos. Siendo esta información vulnerable al no contar con un módulo de seguridad que garantice la protección de datos importantes para la organización. Con estos antecedentes, nace la necesidad del desarrollo de un módulo de seguridad para el sistema del CASI, mismo que presenta como bases las normativas establecidas en las metodologías de seguridad OSSTMM y OWASP.

En este contexto, Voutssas (2010) manifiesta que para poder comprender el concepto integral de la seguridad en los sistemas informáticos, resulta importante comprender diversos conceptos básicos que la rigen para su desarrollo e implementación de la seguridad, pues de otra manera no es

posible establecer una base de investigación. A continuación, se desarrolla las bases teóricas que sustentan el presente estudio:

2.2. Bases teóricas

2.2.1. Open Source Security Testing Methodology Manual (OSSTMM)

OSSTMM es uno de los estándares profesionales más completos y comúnmente utilizados a la hora de revisar la Seguridad de los Sistemas desde Internet (Valdez Alvarado, 2013b). Su metodología se encuentra diseñada para realizar pruebas de seguridad, que se encuentra dividida por secciones, módulos y tareas (Fiaschetti et al., 2015). En cada una de las sección se trata un área que conforma el sistema de información de una organización, además propone la ejecución de tareas con la finalidad de identificar problemas que afecten en gran medida la seguridad de una organización (Sebastian & Parada, 2009). Enfocándose en la protección de la infraestructura de un sistema y todo lo vinculado al mismo.

De esta forma, esta metodología actúa en cinco ámbitos: humano, físico, medios inalámbricos, telecomunicaciones, redes de datos. Sin embargo es importante considerar que OSSTMM aún no es estandarizada, por lo tanto no incluye aplicaciones (Gordón, 2017). No obstante, propone para la optimización de la seguridad de los activos de información, que se disminuyan las limitaciones entre activos de información a proteger y posibles brechas de seguridad, así como también, la no separación de activos de información y brechas de seguridad informática, dando como resultado la porosidad (Gordon & Pacheco, 2018). Además, posee un valor añadido a favor, como lo son las métricas cuantitativas de seguridad Risk Asessment Value (RAV), las mismas que son imprescindibles para gestionar la seguridad informática y de la información dentro de cualquier tipo de organización, sin estas no se podría cuantificar de forma global el estado actual de seguridad Operacional (Gordón, 2017). Convirtiéndose en una medición precisa de la seguridad a nivel operacional, lo cual evita suposiciones y evidencia anecdótica.

Según (Valdez Alvarado, 2013a) su principal propósito es proveer de una metodología científica para examinar la organización, realizando pruebas sobre la seguridad de adentro hacia afuera. Un segundo propósito es proveer guías para el auditor de sistemas, destinadas a la certificación de la organización en cuanto a los requisitos del ISECOM, provee una serie de descripciones específicas para el desarrollo de un test de seguridad operacional sobre todos los canales incluyendo aspectos físicos, humanos, telecomunicaciones, medios inalámbricos, redes de datos y cualquier otra descripción derivada de una métrica real.

Una auditoría de seguridad es el análisis y estudio de un sistema, para intentar conocer sus limitaciones (vulnerabilidades, debilidades, riesgos, etc.) y posteriormente poder corregirlas.

En una auditoría de seguridad es en la que el auditor se centra en localizar las vulnerabilidades del sistema que puedan permitir el acceso de alguien no autorizado. Según la definición de OSSTMM, es un Test de seguridad con un objetivo definido que finaliza cuando el objetivo es alcanzado o el tiempo ha terminado (Sahuquillo, 2011).

La investigación descriptiva sirve para la recopilación de las diversas tendencias y los fundamentos del estudio de la reducción de la vulnerabilidad de seguridad en los sistemas operativos, y la investigación aplicativa permitirá generar criterios sobre la implementación de los diversos procedimientos de detección de errores, para reducir la vulnerabilidad de seguridad en los Sistemas operativos, tomando como fundamento la metodología OSSTMM (Cruz & Martínez, 2017).

2.2.1.1. Módulos y estructura

El flujo de este manual OSSTMM comienza con determinar la situación objetivo, esta situación está determinada por la cultura, reglas, normas, regulaciones, legislación y políticas definidas en esta (Fiaschetti et al., 2015). En este sentido, La metodología propone un modelo jerárquico de Canales, Módulos y Tareas, donde los vectores son simplemente las líneas de análisis que apuntan a cada uno de los canales. Los módulos son áreas específicas de cada canal, pudiendo encontrar actividades que se encuentran en la frontera entre dos canales (Valdez Alvarado, 2013a).

Según manifiesta Miranda (2019) el OSSTMM funciona como una guía, con respecto a los aspectos principales de la seguridad de la información de una empresa, de esta manera permite que el personal autorizado en realizar auditorías, puedan consultar información relevante sobre sus propias políticas de seguridad, esto muestra la gran flexibilidad del manual (Fiaschetti et al., 2015). Las pruebas de seguridad abarcan cinco secciones que corresponden a las secciones que serán de utilidad para el desarrollo de este proyecto. Cada sección consta de varios módulos y cada módulo indica una serie de tareas o pruebas a realizar, cubriendo siguientes áreas.

- Seguridad de la Información
- Seguridad de los Procesos
- Seguridad en las tecnologías de Internet
- Seguridad en las telecomunicaciones
- Seguridad inalámbrica
- Seguridad Física

2.2.1.2. Características

Según Miranda (2019) el OSSTMM cuida de las condiciones y límites tales como los procesos de prueba, ética, análisis de los resultados de las pruebas y la seguridad IT con respecto a la ley, regulaciones y estándares, entre las características se identifican las siguientes:

- Marco legal de soporte
- Es manejable
- Es actual
- Es reconocido
- Económico
- Accesible

En la figura 1-1 del mapa de seguridad OSSTMM se puede observar las secciones identificadas mismas que posteriormente serán utilizadas con la certeza de que cumplirán los objetivos prefijados.

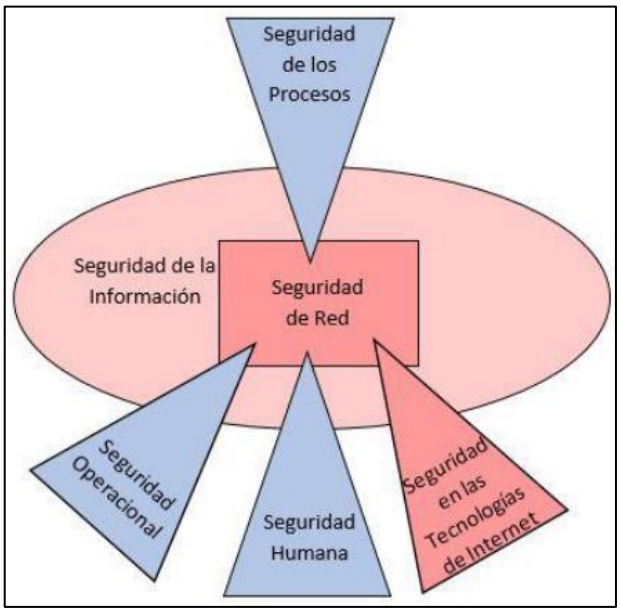


Figura 1-2: Secciones de OSSTMM

Fuente: Miranda (2019).

2.2.1.3. Ámbito de la metodología

El ámbito debe abarcar toda la seguridad operativa, y comprometerse en las diferentes áreas o canales como lo describe el manual (Valdez Alvarado, 2013b), en la tabla 1-2 se puede observar el ámbito de la metodología:

Tabla 1-2: Ámbitos de la metodología

Canal	Sección	Descripción
Seguridad Física	Humano	Hace referencia a todo el personal que se encuentra comprometido con la organización
	Físico	Los espacios y objetos físicos de que forman parte de la organización
Seguridad en comunicaciones	Redes de datos	Sistemas de comunicación y redes de datos.
	Comunicaciones	Comunicaciones y transferencia de datos

Realizado por: Centeno, Hernán, 2024

2.2.1.4. Fases de la metodología

Con base en ISECOM (2012) la metodología OSSTMM, es un documento que reúne de forma estandarizada y ordenada diversas verificaciones y pruebas que se pueden realizar para una auditoría informática.

Esta metodología presenta varias fases, en donde cada una de ellas se asocia con las fases del hacking ético como tipo de prueba que se aplica en este caso de estudio (Gordon & Pacheco, 2018).

Según la perspectiva de Fuertes (2014) OSSTMM presenta una metodología de prueba de intrusión dividida en cuatro fase. Cada una de estas fases aporta una diferente profundidad a la auditoria de seguridad, por lo que no hay ninguna fase más o menos importante dentro de esta metodología (Fiaschetti et al., 2015).

Estas fases se muestran de forma de esquemática en la figura 2.1. Cada fase está compuesta a su vez de diferentes módulos. Las cuatro fases que componen las pruebas de intrusión son los siguientes:

- Preparación
- Interacción
- Investigación
- Intervención

Figura 2-2: Fases de la metodología OSSTMM

Fuente: Fuertes (2014).

2.2.2. Open Web Application Security Project (OWASP)

OWASP es una organización internacional sin fines de lucro dedicada a mejorar la seguridad de las aplicaciones y del software en general (Wichers, 2013). De esta manera, su misión es hacer que la seguridad dentro de las aplicaciones sea más visible para que las organizaciones puedan tomar decisiones sobre conceptos de seguridad basándose en información verídica y contrastada (Toth, 2017). En sí, propone el uso de herramientas automatizadas que permitan la detección de fallos en la configuración o servicios innecesarios, que entorpezcan los procesos y den espacio a fallas en la seguridad de las aplicaciones.

Otro factor que contribuye en las debilidades de seguridad tiene que ver con la falta de actualizaciones de software. Esto incluye el sistema operativo, Servidor Web/Aplicación, DBMS, aplicaciones, y todas las librerías de código (Guerrero et al. 2013).

2.2.2.1. Orientación de OWASP

Entre las diversas facetas de la ciberseguridad, la seguridad del software juega un papel crucial (Burato et al., 2017). Hasta donde sabemos, el OWASP Benchmark Project representa el intento más relevante de establecer un punto de referencia de seguridad universal, es decir, un conjunto de miles de pequeños programas Java que contienen amenazas de seguridad. Según su página web (OWASP, 2016) sus orientaciones son las siguientes:

- **Desarrolladores de Software.** Necesitan usar la guía para asegurarse que el código que se entrega no es vulnerable a ataque. No se puede confiar solo en los testers o grupos de seguridad para que hagan esto por usted. Estos grupos nunca entenderán su aplicación tan bien como usted, y por lo tanto nunca será capaz de probar su aplicación tan efectivamente como usted puede.
- **Testers de Software.** Deberían usar esta guía para mejorar sus habilidades para probar. Mientras las pruebas de seguridad han sido artes oscuras por un largo tiempo, IWASP está trabajando duro para hacer este conocimiento gratuito y abierto para todos. Muchas de las pruebas descritas en esta guía no son complicados y no requieren habilidades especiales o herramientas.
- **Especialistas de Seguridad.** Tiene una responsabilidad especial para asegurar que las aplicaciones no se publiquen con vulnerabilidades. Puede usar esta guía para ayudar a ayudarse a asegurar un nivel de cobertura y rigor. No sea víctima de la trampa de simplemente buscar por algunas vulnerabilidades.

2.2.2.2. Riesgos en seguridad de aplicaciones

La gestión de la seguridad desde el inicio del desarrollo de software evita que los mecanismos de seguridad deban ser ajustados dentro de un diseño ya existente, lo que provocaría cambios que generalmente generan vulnerabilidades en el software, y un incremento de costo y el tiempo para solucionarlos (Niño & Silega, 2018). En tanto que, los atacantes pueden, potencialmente, utilizar diferentes rutas a través de su aplicación para perjudicar su negocio u organización. Cada uno de estos caminos representa un riesgo que puede o no ser suficientemente grave como para merecer atención (Salazar & Silvestre, 2016). Algunas veces, estos caminos son fáciles de encontrar y explotar, mientras que otras son extremadamente difíciles.

De la misma manera, el perjuicio ocasionado puede no tener consecuencias, o puede dejarlo en la quiebra (Niño & Silega, 2018). A fin de determinar el riesgo para su organización, puede evaluar la probabilidad asociada a cada agente de amenaza, vector de ataque, debilidad de seguridad y combinarlo con una estimación del impacto técnico y de negocio para su organización. Juntos, estos factores determinan su riesgo general (Foundation, 2017).

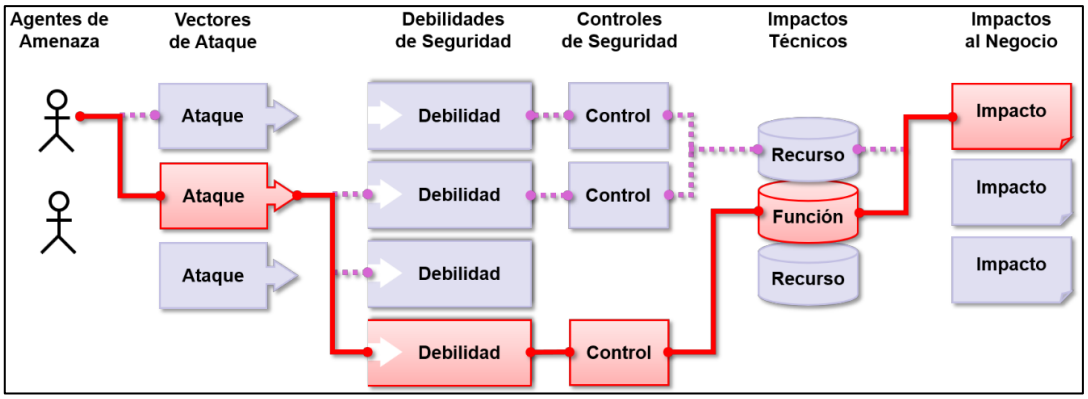


Figura 3-2: Riesgos en seguridad de aplicaciones

Fuente:(Foundation, 2017).

2.2.2.3. OWASP top 10

Según Toth (2017) el OWASP top 10 es uno de los proyectos más populares de OWASP. Es un listado de los problemas de seguridad más comunes encontrados en las aplicaciones web organizados por criticidad, muchas organizaciones lo consideran como un estándar mínimo e incluyen dentro de sus objetivos producir aplicaciones Web que no contengan estas vulnerabilidades (Wichers, 2013). En este sentido, el OWASP Top 10 se enfoca en identificar los riesgos más críticos para un amplio tipo de organizaciones. Para cada uno de estos riesgos, se proporciona información genérica sobre la probabilidad y el impacto técnico, utilizando el siguiente esquema de evaluación, basado en la Metodología de Evaluación de Riesgos de OWASP (Foundation, 2017).

La actualización del OWASP Top se hace en el año 2017, a continuación, se presenta el listado del top 10 de OWASP.

- **A1:2017 Inyección SQL:** Las fallas de inyección, como SQL, NoSQL, OS o LDAP ocurren cuando se envían datos no confiables a un intérprete, como parte de un comando o consulta. Los datos dañinos del atacante pueden engañar al intérprete para que ejecute comandos involuntarios o acceda a los datos sin la debida autorización.
- **A2:2017 Pérdida de Autenticación:** Las funciones de la aplicación relacionadas a autenticación y gestión de sesiones son implementadas incorrectamente, permitiendo a los atacantes comprometer usuarios y contraseñas, token de sesiones, o explotar otras fallas de implementación para asumir la identidad de otros usuarios (temporal o permanentemente).
- **A3:201 Exposición de datos sensibles:** Muchas aplicaciones web y APIs no protegen adecuadamente datos sensibles, tales como información financiera, de salud o Información Personalmente Identificable (PII). Los atacantes pueden robar o modificar estos datos

protegidos inadecuadamente para llevar a cabo fraudes con tarjetas de crédito, robos de identidad u otros delitos. Los datos sensibles requieren métodos de protección adicionales, como el cifrado en almacenamiento y tránsito.

- **A4:2017 Entidades Externas XML (XXE):** Muchos procesadores XML antiguos o mal configurados evalúan referencias a entidades externas en documentos XML. Las entidades externas pueden utilizarse para revelar archivos internos mediante la URI o archivos internos en servidores no actualizados, escanear puertos de la LAN, ejecutar código de forma remota y realizar ataques de denegación de servicio (DoS).
- **A5:2017 Pérdida de Control de Acceso:** Las restricciones sobre lo que los usuarios autenticados pueden hacer no se aplican correctamente. Los atacantes pueden explotar estos defectos para acceder, de forma no autorizada, a funcionalidades y/o datos, cuentas de otros usuarios, ver archivos sensibles, modificar datos, cambiar derechos de acceso y permisos, etc.
- **A6:2017 Configuración de Seguridad Incorrecta:** La configuración de seguridad incorrecta es un problema muy común y se debe en parte a establecer la configuración de forma manual, ad hoc o por omisión (o directamente por la falta de configuración). Son ejemplos: S3 buckets abiertos, cabeceras HTTP mal configuradas, mensajes de error con contenido sensible, falta de parches y actualizaciones, frameworks, dependencias y componentes desactualizados, etc.
- **A7:2017 Secuencia de Comandos en Sitios Cruzados (XSS):** Los XSS ocurren cuando una aplicación toma datos no confiables y los envía al navegador web sin una validación y codificación apropiada; o actualiza una página web existente con datos suministrados por el usuario utilizando una API que ejecuta JavaScript en el navegador. Permiten ejecutar comandos en el navegador de la víctima y el atacante puede secuestrar una sesión, modificar (defacement) los sitios web, o redireccionar al usuario hacia un sitio malicioso.
- **A8:2017 Deserialización Insegura:** Estos defectos ocurren cuando una aplicación recibe objetos serializados dañinos y estos objetos pueden ser manipulados o borrados por el atacante para realizar ataques de repetición, inyecciones o elevar sus privilegios de ejecución. En el peor de los casos, la deserialización insegura puede conducir a la ejecución remota de código en el servidor.
- **A9:2017 Componentes con vulnerabilidades conocidas:** Los componentes como bibliotecas, frameworks y otros módulos se ejecutan con los mismos privilegios que la aplicación. Si se explota un componente vulnerable, el ataque puede provocar una pérdida de datos o tomar el control del servidor. Las aplicaciones y API que utilizan componentes con vulnerabilidades conocidas pueden debilitar las defensas de las aplicaciones y permitir diversos ataques e impactos.

- **A10:2017 Registro y Monitoreo Insuficientes:** El registro y monitoreo insuficiente, junto a la falta de respuesta ante incidentes permiten a los atacantes mantener el ataque en el tiempo, pivotear a otros sistemas y manipular, extraer o destruir datos. Los estudios muestran que el tiempo de detección de una brecha de seguridad es mayor a 200 días, siendo típicamente detectado por terceros en lugar de por procesos internos.

2.2.2.4. Aplicaciones web

La World Wide Web (WWW) o, de forma más coloquial, la web, se ha convertido, junto con el correo electrónico, en el principal caballo de batalla de Internet (Berners et al., 1994). Ésta ha dejado de ser una inmensa "biblioteca" de páginas estáticas para convertirse en un servicio que permite acceder a multitud de prestaciones y funciones, así como a infinidad de servicios, programas, tiendas, etc (Carles, 2013). Es importante indicar que a pesar de su papel cada vez más importante en la comunicación, la www sigue sin estar controlada: cualquier individuo o institución puede crear un sitio web con cualquier número de documentos y enlaces.

Las aplicaciones web se han convertido en pocos años en complejos sistemas con interfaces de usuario cada vez más parecidas a las aplicaciones de escritorio, dando servicio a procesos de negocio de considerable envergadura y estableciéndose sobre ellas requisitos estrictos de accesibilidad y respuesta (Carles, 2013). Esto ha exigido reflexiones sobre la mejor arquitectura y las técnicas de diseño más adecuadas. En este artículo se pretende dar un breve repaso a la arquitectura de tales aplicaciones y a los patrones de diseño más aplicables (Garrido, 2014).

En el desarrollo de aplicaciones web se presentan gran variedad de herramientas que reducen el trabajo y que cuentan con características particulares que facilitan la construcción en algunos tipos de aplicaciones, con el objetivo de optimizar la construcción de una aplicación web, surge el siguiente trabajo que ofrece una comparación de los frameworks de los lenguajes HTML5, CSS y JavaScript, evaluando características, ventajas, desventajas y limitaciones (Valbuena, 2014).

En el desarrollo de aplicaciones informáticas centradas en el usuario, la usabilidad va apareciendo como un método de desarrollo de aplicaciones basadas en Web, con un rol más importante si cabe que la propia arquitectura de información o la gestión de contenidos (Piattini et al., 2020). Esto constituye un aliciente que ha despertado gran interés en torno a la integración y evaluación de la usabilidad en el proceso de diseño e implementación, así como en la aplicación en uso (Caballero, 2007). Por esta razón, el interés por evaluar la usabilidad de sitios Web está aumentando rápidamente (Alva, 2005). El usuario interacciona con las aplicaciones web a través del navegador.

Como consecuencia de la actividad del usuario, se envían peticiones al servidor, donde se aloja la aplicación y que normalmente hace uso de una base de datos que almacena toda la información relacionada con la misma. El servidor procesa la petición y devuelve la respuesta al navegador que la presenta al usuario. Por tanto, el sistema se distribuye en tres componentes: el navegador, que presenta la interfaz al usuario; la aplicación, que se encarga de realizar las operaciones necesarias según las acciones llevadas a cabo por éste y la base de datos, donde la información relacionada con la aplicación se hace persistente. Esta distribución se conoce como el modelo o arquitectura de tres capas (Garrido, 2014).

El diseño de un sistema web es el proceso que extiende, refina y reorganiza los aspectos detectados en el proceso de modelado conceptual para generar una especificación rigurosa del sistema de información siempre orientada a la obtención de la solución del sistema software (Jiménez, 2012). Un diseño de sitios web se basa en una arquitectura de tres capas (presentación, aplicación y persistencia). La utilización de esta arquitectura se debe a que los distintos niveles son independientes unos de otros de manera que, por ejemplo, se puede cambiar fácilmente el comportamiento de las clases en el nivel de aplicación sin que ello influya en las otras capas (Pérez, 2015).

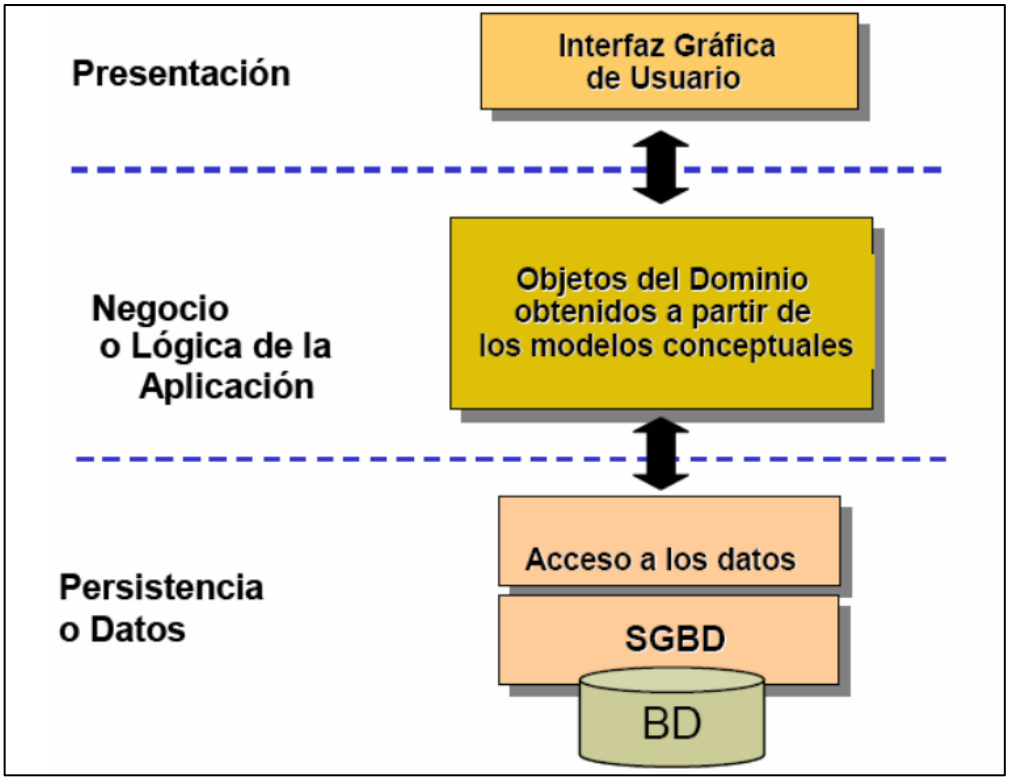


Figura 4-2: Esquema de tres capas de sistemas web

Fuente: Pérez (2015)

2.2.2.5. Protocolo HTTP

El Protocolo de Transferencia de HiperTexto (Hypertext Transfer Protocol) es un sencillo protocolo cliente-servidor que articula los intercambios de información entre los clientes Web y los servidores HTTP (Avogadro, 2007).

La especificación completa del protocolo HTTP 1/1 está recogida en el RFC 2616. Fue propuesto por Tim Berners-Lee, atendiendo a las necesidades de un sistema global de distribución de información como el World Wide Web (Vazques, 2017). El desarrollo de aplicaciones web necesita frecuentemente mantener estado. Para esto se usan las cookies, que es información que un servidor puede almacenar en el sistema cliente.

Desde el punto de vista de las comunicaciones, está soportado sobre los servicios de conexión TCP/IP: un proceso servidor escucha en un puerto de comunicaciones TCP (por defecto, el 80), y espera las solicitudes de conexión de los clientes Web (Avogadro, 2007). Una vez que se establece la conexión, el protocolo TCP se encarga de mantener la comunicación y garantizar un intercambio de datos libre de errores.

HTTP se basa en sencillas operaciones de solicitud/respuesta (Meyer, 2004). Un cliente establece una conexión con un servidor y envía un mensaje con los datos de la solicitud. El servidor responde con un mensaje similar, que contiene el estado de la operación y su posible resultado. Todas las operaciones pueden adjuntar un objeto o recurso sobre el que actúan; cada objeto Web es conocido por su URL (Vazques, 2017).

En una aplicación web, las vistas serían las páginas HTML que el usuario visualiza en el navegador (Humpleman et al., 2001). A través de estas páginas el usuario interactúa con la aplicación, enviando eventos al servidor a través de peticiones HTTP.

En el servidor se encuentra el código de control para estos eventos, que en función del evento concreto actúa sobre el modelo convenientemente. Los resultados de la acción se devuelven al usuario en forma de página HTML mediante la respuesta HTTP (Garrido, 2014).

Según Prieto (2014) el diálogo con los servidores HTTP se establece a través de mensajes formados por líneas de texto, cada una de las cuales contiene los diferentes comandos y opciones del protocolo. Solo existen dos tipos de mensajes, uno para realizar peticiones y otro para devolver la correspondiente respuesta (Humpleman et al., 2001). La estructura general de los dos tipos de mensajes se puede ver en el siguiente esquema:

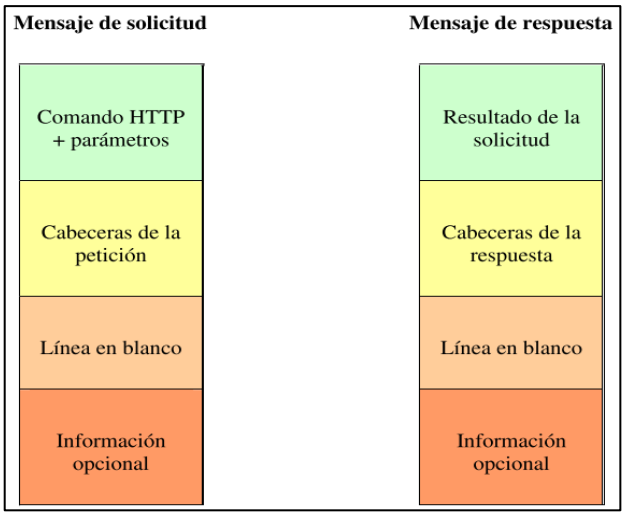


Figura 5-2: Estructura de HTTP

Fuente: Prieto (2014)

2.2.2.6. Servidor web

Los documentos Web o también llamados páginas Web pueden estar localizados en diferentes sitios de Internet, estos sitios son llamados servidores Web. De manera que un documento WWW puede contener enlaces a otros documentos que se encuentran en el mismo servidor Web o en otros servidores Web, logrando así formar una telaraña mundial de información (Murcia, 2011).

Se trata de un programa que implementa el protocolo HTTP. Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas web o páginas HTML: textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música (Pérez, 2015).

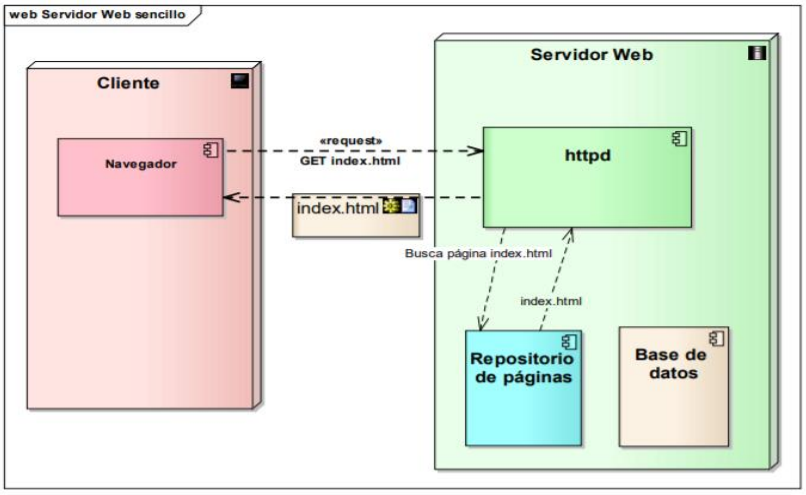


Figura 6-2: Esquema de funcionamiento de un servidor web

Fuente: Mestras (2013)

Según Montenegro (2016) un servidor puede ofrecer distintos tipos de servicios. Cada uno de ellos se realiza a través de un puerto en la máquina servidor, cada puerto está identificado con un número entre 0 y 65535. En la Figura 7-1 se puede observar los principales puertos que dispone un servidor.

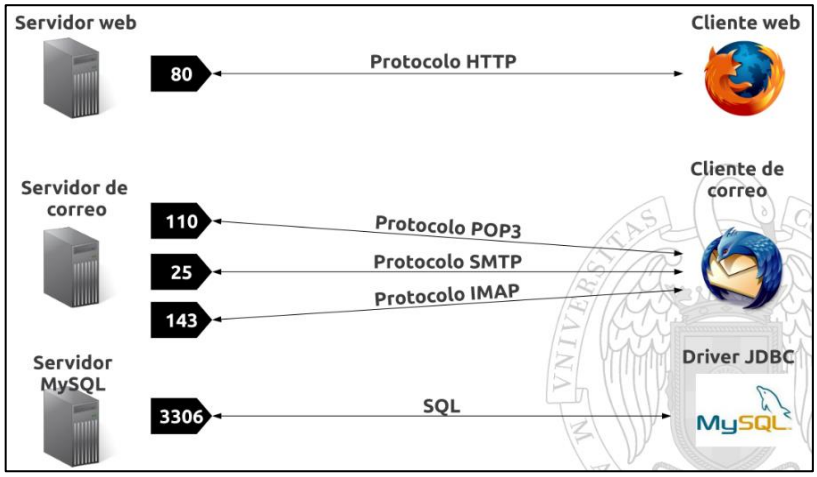


Figura 7-2: Puerto de los servidores

Fuente: Montenegro (2016)

Entre los servidores web más utilizados se describe los siguientes:

- **Apache:** Es el servidor más utilizado, aunque ha vivido tiempos mejores. Parte de su éxito se debe a que es multiplataforma y a su estructura modular, que permite emplear diversos lenguajes en el lado del servidor (PHP, Python y Perl principalmente), así como incorporar características como la compresión de datos, las conexiones seguras y la utilización de URLs amigables (Chavarria & Gudiño, 2017).
- **Microsoft IIS:** Internet Information Server (IIS) es un servidor Web que permite publicar información en una intranet o en Internet. Este servidor Web se apoya sobre el protocolo de transferencia de hipertexto (HTTP), para transmitir la información y comunicarse con los clientes, la configuración de IIS se basa en la utilización de hojas de propiedades. Cada hoja de propiedades mostrará una serie de parámetros para configurar, dependiendo del elemento seleccionado (directorio, sitio Web o fichero) (Mejías, 2018).

2.2.2.7. Arquitectura cliente servidor

El modelo Cliente/Servidor permite diversificar el trabajo que realiza cada aplicación, de forma que los Clientes no se sobrecarguen, cosa que ocurriría si ellos mismos desempeñan las funciones que le son proporcionadas de forma directa y transparente (Gimenez et al., 2016). En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más

importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema. Tanto el Cliente como el Servidor son entidades abstractas que pueden residir en la misma máquina o en máquinas diferentes (Marini, 2014).

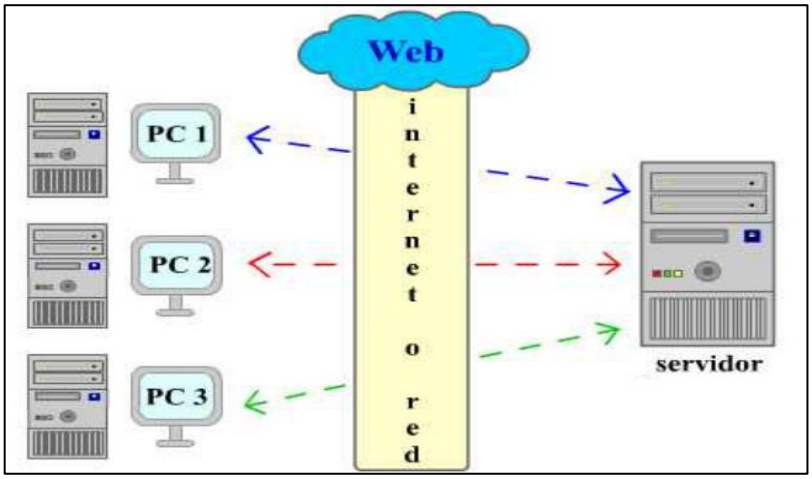


Figura 8-2: Esquema de funcionamiento de un servidor web

Fuente: Infante (2014)

2.2.3. Angular JS

El término framework, hace referencia a una estructura de software compuesta por componentes personalizables e intercambiables para el desarrollo de una aplicación (Holsti, 1972). En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que se le puede añadir las últimas piezas para construir una aplicación concreta (Valbuena, 2014).

Angular.js es un framework JavaScript Open Source desarrollado por Google, sirve para crear Webapps en lenguaje cliente con JavaScript ejecutándose con el conocido single-page applications (aplicación de una sola página) que extiende el tradicional HTML con etiquetas propias (Steyer & Softic, 2016). Es totalmente extensible y funciona bien con otras bibliotecas (Valbuena, 2014). Una de las ventajas que presenta este framework es el Data-Binding es decir que, si en el backend cambia el valor de una variable, se verá reflejado en el frontend. Si el cambio surge en una variable del frontend, el valor de la variable automáticamente se actualizará sin tener que hacer llamadas extras (Cangás, 2015).

AngularJS es un framework de código abierto en JavaScript, creado y soportado por Google, contiene un conjunto de librerías útiles para el desarrollo de aplicaciones web del lado del cliente,

representa la "A" del stack MEAN (Azaustre, 2014). En las aplicaciones web actuales, el front-end se encuentra libre de incrustaciones de código back-end (PHP-Java), separando su funcionalidad de las otras capas, lo realiza usando plantillas con algunas etiquetas incrustadas que hacen las llamadas o bindings28 hacia el back-end (Cangás, 2015).

2.2.3.1. Node JS

Node.js es un marco de E / S asíncrono basado en eventos, sin bloqueo, que utiliza el motor de JavaScript V8 de Google (Cantelon et al., 2014). Se utiliza para desarrollar aplicaciones que hacen un uso intensivo de la capacidad de ejecutar JavaScript tanto en el cliente como en el lado del servidor y, por lo tanto, se benefician de la reutilización del código y la falta de cambio de contexto. Es de código abierto y multiplataforma. Las aplicaciones Node.js están escritas en JavaScript puro y se pueden ejecutar dentro del entorno Node.js en Windows, Linux, etc (Node.js, 2017).

2.2.3.2. JavaScript del Lado del Servidor

Las primeras encarnaciones de JavaScript vivían en los browsers (Flanagan & Matilainen, 2007). Pero esto es sólo el contexto. Define lo que puedes hacer con el lenguaje, pero no dice mucho acerca de lo que el lenguaje mismo puede hacer. JavaScript es un lenguaje "completo": Lo puedes usar en muchos contextos y alcanzar con éste, todo lo que puedes alcanzar con cualquier otro lenguaje "completo" (Kiessling & Junge, 2015).

Node.js realmente es sólo otro contexto: te permite correr código JavaScript en el backend, fuera del browser. Para ejecutar el código JavaScript que tu pretendes correr en el backend, este necesita ser interpretado y, bueno, ejecutado, Esto es lo que Node.js realiza, haciendo uso de la Máquina Virtual V8 de Google, el mismo entorno de ejecución para JavaScript que Google Chrome utiliza (Kiessling & Junge, 2015). Además, Node.js viene con muchos módulos útiles, de manera que no tienes que escribir todo de cero, como, por ejemplo, algo que ponga un string a la consola. Entonces, Node.js es en realidad dos cosas: un entorno de ejecución y una librería (Kiessling & Junge, 2015).

2.2.3.3. Servidor web web Node

Escribir aplicaciones Node.js es una cosa, entender por qué ellas necesitan ser escritas en la manera que lo son significa entender JavaScript (Arias, 2017). JavaScript realmente vive dos, o tal vez tres vidas (El pequeño ayudante DHTML de mediados de los 90's, las cosas más serias tales como jQuery y similares, y ahora, el lado del servidor), no es tan fácil encontrar información que

te ayude a aprender JavaScript de la "manera correcta", de forma de poder escribir aplicaciones de Node.js en una apariencia que te haga sentir que no sólo estás usando JavaScript (Kiessling & Junge, 2015).

Un servidor Node puede realizar un procesamiento de una petición, según (Curso, 2017) en cada petición de recurso se ejecuta la función pasada a créate Server y el primer parámetro de la función manejadora permite el acceso a la petición HTTP.

2.2.3.4. Typescript

TypeScript admite interfaces, pero el compilador genera JavaScript, que no lo hace. Por lo tanto, las interfaces se pierden efectivamente en el paso de compilación. Esta es la razón por la que la verificación de tipos en las interfaces se basa en la forma del objeto, es decir, si el objeto es compatible con los campos y funciones de la interfaz, y no en si la interfaz se implementa o no (TypeScript, 2017).

Según Valverde & Mora (2017) TypeScript es un lenguaje precompilado, es decir, un lenguaje el cual será compilado finalmengte a javascript, la versión del javascript en la cual será compilado junto con otras configuraciones estará en el archivo tsconfig, TypeScript nos proporciona una serie de ventajas sobre javascript, ya que tiene una serie de características como por ejemplo: interfaces, clases (clases de verdad) y es fuertemente tipado.

2.2.3.5. Características de Angular JS:

Según Cangás (2015) entre las características esenciales que Angular JS presenta son las siguientes:

- **Data-Binding:** En las aplicaciones web, si en el backend cambia el valor de una variable, se verá reflejado en el frontend. Si el cambio surge en una variable del frontend, el valor de la variable automáticamente se actualizará sin tener que hacer llamadas extras.
- **MVC (Modelo Vista Controlador):** La estructura de AngularJS, obliga de manera ortodoxa a tener el código ordenado, haciendo uso de controladores para enviar datos a la vista, modelos para base de datos, directivas, servicios o filtros, para la inyección de dependencias.

El modelo Cliente/Servidor permite diversificar el trabajo que realiza cada aplicación, de forma que los Clientes no se sobrecarguen, cosa que ocurriría si ellos mismos desempeñan las funciones que le son proporcionadas de forma directa y transparente (Orfali et al., 2002). En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más

importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema. Tanto el Cliente como el Servidor son entidades abstractas que pueden residir en la misma máquina o en máquinas diferentes (Marini, 2014).

El Modelo es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico de los Controladores o de las Vistas, ni siquiera contiene referencias a ellos. Es el propio sistema el que tiene encomendada la responsabilidad de mantener enlaces entre el gModelo y sus Vistas, y notificar a las Vistas cuando cambia el Modelo (Fernández & Díaz, 2012).

La Vista es el objeto que maneja la presentación visual de los datos representados por el Modelo. Genera una representación visual del Modelo y muestra los datos al usuario. Interactúa preferentemente con el Controlador, pero es posible que trate directamente con el Modelo a través de una referencia al propio Modelo (Fernández & Díaz, 2012).

El Controlador es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el Modelo, centra toda la interacción entre la Vista y el Modelo. Cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del Modelo o por alteraciones de la Vista. Interactúa con el Modelo a través de una referencia al propio Modelo (Fernández & Díaz, 2012).

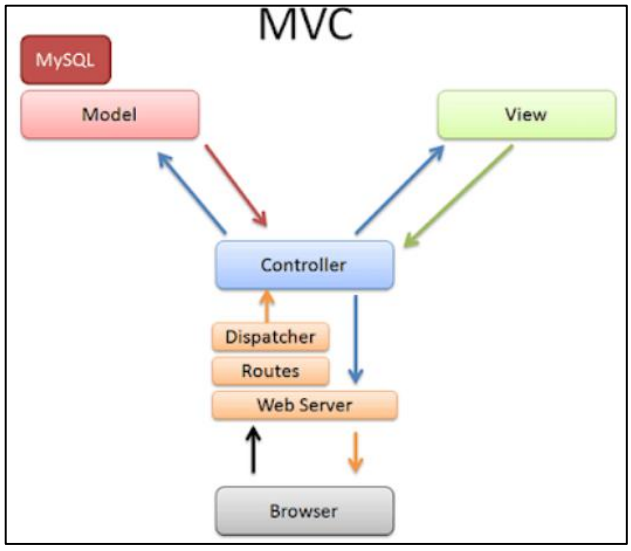


Figura 9-2: Elementos del MVC

Fuente: Tapuy (2019)

2.2.4. Servicios JavaScript Object Notation (Json)

Un servicio web según Axis (2015) es un componente de software reutilizable y distribuido que ofrece una funcionalidad concreta, independiente tanto del lenguaje de programación en que está implementado como de la plataforma de ejecución.

La seguridad del Servicio Web estará a cargo de JSON Web Token. Se deberá definir un recurso específico para la generación de tokens de acceso. Este recurso comprobará que en la llamada al mismo se incorpore la clave de seguridad necesaria para la obtención del token, si ésta es correcta se devolverá al usuario un token de acceso codificado según el modelo de JSON Web Token a través de claims. Una vez se tenga este token de acceso, el usuario deberá incorporarlo a la cabecera de las peticiones HTTP que se realicen a cada uno de los recursos de obtención de datos definidos en el primer punto. En cada uno de los recursos se validará que sea un token válido y que éste no haya expirado (Monago, 2017).

2.2.4.1. Peticiones a un servicio JSON

Según Rodríguez (2016) entre las peticiones a un servicio JSON están las siguientes:

- **GET.** El método GET es usado para leer la representación de un recurso, sin modificarlo o dañarlo. Una respuesta exitosa generalmente incluye una representación en XML o JSON y el código 200 para OK. En caso de error retorna el código 404 (not found) o 400 (bad request). GET es una operación segura e idempotente.
- **PUT.** PUT se usa para actualizar recursos. En el cuerpo del mensaje se envían los datos que actualizarán el recurso original. Algunos servicios usan PUT para crear un nuevo recurso cuando la ID enviada por el cliente no corresponde a un recurso existente, aunque esto no es recomendable porque genera confusión con el verbo POST. El cuerpo en la respuesta del PUT es opcional. PUT es idempotente, pero no es seguro porque modifica el estado del recurso.
- **POST.** Es el verbo usado para crear recursos subordinados. Es decir, que al nuevo recurso se le asigna una ID que lo asocia con el padre. La respuesta retorna el código 201 para indicar una creación exitosa. POST no es idempotente ni seguro, porque envíos sucesivos de la misma solicitud pueden generar la creación de sendos recursos.
- **DELETE.** Usado para eliminar un recurso identificado por su URI. Un borrado exitoso retorna el código 200 con el cuerpo de la respuesta, o 204 si no se incluye contenido en el cuerpo de la respuesta. DELETE no siempre puede considerarse idempotente. Por ejemplo, si se trata de una base de datos, se podría recibir el error 404 (not found) en lugar de marcar el recurso como borrado.

2.2.4.2. Ventajas de los servicios

El uso de servicios REST FULL sobre HTTP/JSON provee la estandarización de los argumentos, los formatos de datos y el mecanismo de transporte de las interfaces del sistema de gestión propuesto (Fajardo & Silva, 2014). Si se toma en cuenta que la serialización/deserialización de objetos es un mecanismo por el cual un objeto pasa de ser objeto a texto, es transportado a cualquier destino y finalmente se vuelve a transformar a objeto, dicha información teórica resulta importante, ya que enriquecería más el tema para los usuarios de JSON en el desarrollo de sus aplicaciones (Mora, 2016).

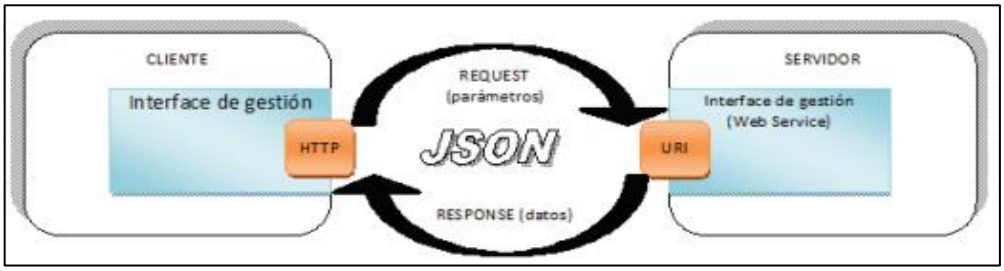


Figura 10-2: Esquema de servicio JSON

Fuente: (Fajardo & Silva, 2014)

2.2.5. Ataques y seguridad

La falta de seguridad informática hoy en día es una latente preocupación en el campo de las redes especialmente donde se maneja información confidencial, el presente artículo tiene como finalidad el análisis y detección de vulnerabilidades. El aumento de la interconectividad informática y la popularidad del Internet están ofreciendo a las organizaciones todo tipo de oportunidades sin precedentes para mejorar las operaciones, reduciendo significativamente el uso del papel, a su vez reduciendo los costos al compartir información. Sin embargo, el éxito de muchos de estos esfuerzos depende, en gran parte, de la capacidad de la organización para proteger la integridad, confidencialidad y disponibilidad de los datos y de los sistemas informáticos. Aunque la seguridad de la información juega un papel importante en la protección de los datos y de los activos de una organización, a menudo oímos noticias sobre delitos informáticos, como la alteración de sitios web o robo de datos (Miranda, 2019).

La falta de seguridad informática hoy en día es una latente preocupación en el campo de las redes especialmente donde se maneja información confidencial, el presente artículo tiene como finalidad el análisis y detección de vulnerabilidades. El aumento de la interconectividad informática y la popularidad del Internet están ofreciendo a las organizaciones todo tipo de

oportunidades sin precedentes para mejorar las operaciones, reduciendo significativamente el uso del papel, a su vez reduciendo los costos compartir información. Sin embargo, el éxito de muchos de estos esfuerzos depende, en gran parte, de la capacidad de la organización para proteger la integridad, confidencialidad y disponibilidad de los datos y de los sistemas informáticos. Aunque la seguridad de la información juega un papel importante en la protección de los datos y de los activos de una organización, a menudo oímos noticias sobre delitos informáticos, como la alteración de sitios web o robo de datos (Miranda, 2019).

Cada día se descubren nuevos puntos débiles y, por lo general, son pocos los responsables de IT que comprenden en su justa medida la importancia que tiene la seguridad y cómo pueden abordar el grave problema que existe detrás de vulnerabilidades que permiten a un atacante, violar la seguridad de un entorno y cometer delitos en función de los datos robados (Mieres, 2009).

2.2.5.1. Fases de un ataque informático

El pensar como un atacante en la parte de seguridad informática nos mantiene un paso delante de ellos, pero jamás se debe subestimar su mentalidad, para eso se debe conocer las fases de un ataque informático. La siguiente imagen muestra las cinco etapas por las cuales suele pasar un ataque informático al momento de ser ejecutado (Alvear, 2019).

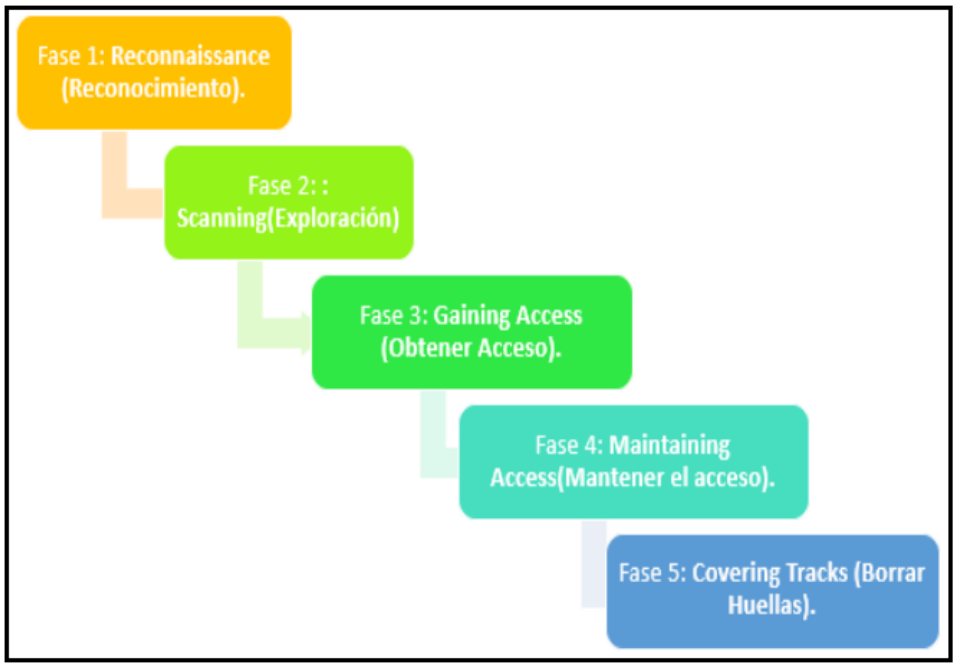


Figura 11-2: Fases de un ataque informático

Fuente: (Alvear, 2019)

2.2.5.2. Ethical hacking

Es una metodología utilizada para simular un ataque malicioso sin causar daño con el fin de analizar las brechas de seguridad que contiene una red , esta práctica tiene como fin poder identificar los riesgos a los que se encuentran expuesta la red de una organización. El ethical hacking se basa en procedimientos basados en una investigación preliminar de análisis de vulnerabilidades, donde la acción inicia en el momento de identificar vulnerabilidades críticas del sistema, después de realizar la identificación se realiza la explotación de vulnerabilidades, donde se evidencia los puntos sensibles que se encuentran expuestos, se procede a realizar un análisis con el fin de dar las respectivas recomendaciones de como mitigar las brechas de seguridad encontradas (Cañón, 2015).

2.2.5.3. Cyber ataques

Según el Centro Criptológico Nacional (2019) los ciberdelincuentes continúan siendo uno de los grupos de agentes de las amenazas más activos, con más del 80% de la actividad dañina.

La Ciberdelincuencia como fenómeno complejo y global, requiere un enfoque multidisciplinar para abordar cualquier planteamiento de respuesta contra la misma. Para ello, una primera aproximación impone el conocimiento y la visualización de la realidad criminal a la que nos enfrentamos. El conocimiento de esta realidad viene obligado a describir aspectos no solamente relacionados con los datos estadísticos, sino que implica también ahondar en otras temáticas de referencia que deben ser consignadas aquí para dimensionar y comprender adecuadamente el fenómeno de la ciberdelincuencia (Fernandes et al., 2019).

La denominada ciberdelincuencia se ha expandido sustentada en el desarrollo de las nuevas tecnologías, espacio sin fronteras bien establecidas en el cual se pueden cometer una serie de actos vandálicos sin las restricciones que se imponen en el entorno tradicional. Esto ha exigido el surgimiento de una sociedad de la información que se enfoque en la prevención, la investigación, la prueba y la represión del hecho criminal superando las trabas que presentan las jurisdicciones de los distintos países (Fernández et al., 2019).

2.2.5.4. Ransomware

El negocio del ransomware existe porque hay un mercado cada vez más demandante del servicio. Por un lado, los ciberdelincuentes ganan dinero distribuyendo sus códigos maliciosos y cobrando los correspondientes rescates. Por el otro, ganan dinero vendiendo sus propios ransomware para

que un tercero pueda utilizarlos, de la misma manera que una empresa legítima de software vende sus productos. Un tercer modelo implica que el proveedor del servicio de ransomware cobra a sus clientes un porcentaje de los rescates ($) pagados por las víctimas (OWASP, 2018).

El ransomware, a pesar de ser una de las fuentes más rentables de ingresos para los ciberdelincuentes, ha disminuido para observar un repunte muy importante en malware dedicado a blockchain. El crecimiento en 2017 del valor del bitcoin ha hecho redirigir esfuerzos a este tipo de amenazas. No nos engañemos, el consumo de proceso y los fallos en usabilidad de los sitios web comprometidos suponen también un importante impacto económico para las empresas (Datacom, 2019).

2.2.5.5. Troyano

Troyanos: también llamados caballos de Troya son un programa malicioso que se esconde dentro de otro programa no maligno, y que al ejecutarse este último el código malicioso también se ejecuta, siendo transparente para el usuario. El troyano hace que el atacante tenga acceso remoto a la información y recursos de la víctima, permitiéndole robar o dañar información del equipo (Sebastian & Parada, 2009).

Según Fernandes et al. (2019) un troyano de puerta trasera / acceso remoto (RAT) accede a un sistema informático o dispositivo móvil de forma remota. Puede ser instalado por otra pieza de malware. Le da un control casi total al atacante, que puede realizar una amplia gama de acciones, que incluyen:

- Acciones de monitoreo
- Ejecutar comandos
- Enviando archivos y documentos al atacante
- Pulsaciones de teclas de registro
- Tomar capturas de pantalla

2.2.5.6. Inyecciones SQL

Consiste en la inserción de código SQL por medio de los datos de entrada desde la parte del cliente hacia la aplicación. Es decir, por medio de la inserción de este código el atacante puede modificar las consultar originales que debe realizar la aplicación y ejecutar otras totalmente distintas con la intención de acceder a la herramienta, obtener información de alguna de las tablas o borrar los datos almacenados, entre otras muchas cosas (Cardenal, 2013).

La forma más sencilla de un ataque base de las SQL inyección es mostrar los datos en pantalla para el atacante, pero en ocasiones este ataque no funciona y se requiere de otra condición por el cual la información o datos sean obtenidos, para este caso se hace uso de ataques a inyecciones a ciegas(blind Sql injection), donde su objetivo radica en el momento de inyectar código y obtener la información de la base de datos, basado en el comportamiento de la aplicación, es decir, el procedimiento es detectar el ingreso de un sentencia o parámetro vulnerable y observar el resultado o respuesta obtenida de la aplicación (Chalabe, 2019).

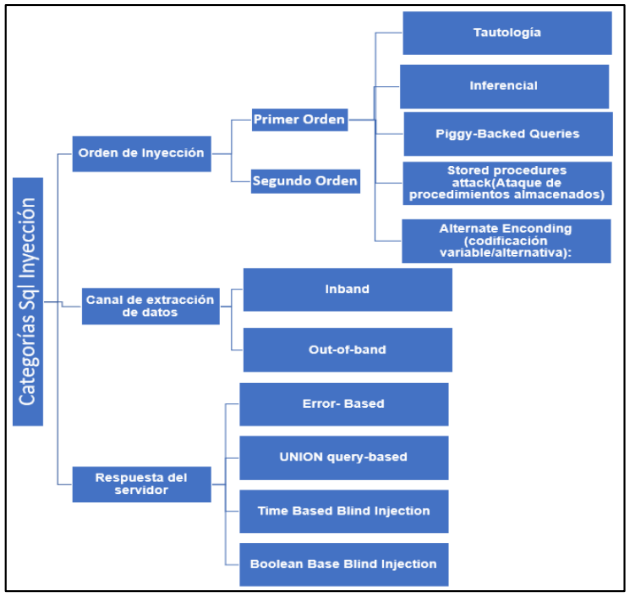


Figura 12-2: Categorías de SQL Inyection

Fuente: Chalabe (2019)

2.2.5.7. Principales problemas que causan los ataques SQL Injection

Entre los principales problemas que causal los ataques SQL Injection según manifiesta Cardenal (2013) se encuentran los siguientes:

- **Confidencialidad.** De forma habitual, las bases de datos almacenan información sensible, por lo que la pérdida de confiabilidad es un problema muy frecuente en aquellos sitios que son vulnerables a este tipo de ataques.
- **Autenticación.** Si el sistema de logueo que se utiliza para acceder a una zona restringida de una web es débil, por medio de este tipo de ataques se podría acceder sin la necesidad de conocer ni el usuario ni la contraseña.
- **Integridad.** Al igual que un ataque por inyección SQL permite leer información relevante almacenada en la base de datos, también es posible realizar cambios o incluso borrar toda información mediante este tipo de vulnerabilidad.

2.2.5.8. Acciones que se puede realizar mediante una inyección SQL

Mediante la inyección SQL un atacante podría realizar entre otras cosas las siguientes acciones contra el sistema según Cebrián et al. (2015):

- **Descubrimiento de información (information disclosure):** Las técnicas de inyección SQL pueden permitir a un atacante modificar consultas para acceder a registros y/o objetos de la base de datos a los que inicialmente no tenía acceso.
- **Elevación de privilegios:** Todos los sistemas de autenticación que utilicen credenciales almacenados en motores de bases de datos hacen que una vulnerabilidad de inyección SQL pueda permitir a un atacante acceder a los identificadores de usuarios más privilegiados y cambiarse las credenciales.
- **Denegación de servicio:** La modificación de comandos SQL puede llevar a la ejecución de acciones destructivas como el borrado de datos, objetos o la parada de servicios con comandos de parada y arranque de los sistemas. Asimismo, se pueden inyectar comandos que generen un alto cómputo en el motor de base de datos que haga que el servicio no responda en tiempos útiles a los usuarios legales.
- **Suplantación de usuarios:** Al poder acceder al sistema de credenciales, es posible que un atacante obtenga las credenciales de otro usuario y realice acciones con la identidad robada o "spoofeada" a otro usuario.

2.2.6. Seguridad informática

El término seguridad proviene del latín "securitas", éste significa el tener conocimiento y certeza sobre algo. La palabra seguridad refiere a la ausencia del peligro, miedo y riesgos (Miranda, 2019). Actualmente la informática y en especial la información es uno de los activos principales de las organizaciones y empresas, existen diferentes tipos de amenazas que atentan contra el buen funcionamiento de estos entes, como los virus, los malware, cibercriminales, spyware y un sinnúmero de amenazas existentes, diariamente se utilizan diferentes equipos en especial móviles que están conectados a internet, la mayor fuente de amenazas para la seguridad (Romero et al., 2018).

El proceso de establecer y observar un conjunto de estrategias, políticas, técnicas, reglas, guías, prácticas y procedimientos tendientes a prevenir, proteger y resguardar de daño, alteración o sustracción a los recursos informáticos de una organización y que administren el riesgo al garantizar en la mayor medida posible el correcto funcionamiento ininterrumpido de esos recursos (Quiroz & Macías, 2017). La ciberseguridad ha progresado de forma eficiente en el sector energético debido a la trascendencia de las infraestructuras críticas para los servicios públicos, el alto valor

de los activos empresariales a proteger y, desafortunadamente, por la necesidad de defenderse ante los ciberataques que tienen al sector en su punto de mira (Arteaga, 2019).

Al hablar de términos de seguridad informática se debe entender a las bases que conforman los cimientos de esta ciencia, para las partes más complejas de esta disciplina, una de estas bases es el concepto de seguridad, la cual consiste en un estado de bienestar, es la ausencia de riesgo por la confianza que existe en alguien o algo, si la seguridad se aborda desde el tema disciplinario el concepto se puede definir como una ciencia interdisciplinaria para evaluar y gestionar los riesgos a los que se encuentra una persona, un animal, el ambiente o un bien. Existen países en donde la seguridad es un tema nacional, aunque depende del tipo de seguridad, existen muchos tipos de ésta, por ejemplo, la seguridad ambiental, la seguridad económica, la seguridad sanitaria y en casi la mayoría de los países cuando se hace un análisis de la palabra seguridad, se hace referencia a la seguridad de las personas, por ejemplo, evitar el estado de riesgo de un robo, de un daño físico o de un bien material (Romero Castro et al., 2018).

Según Alvear (2019) en la seguridad informática existen cuatro elementos fundamentales que son el mayor objetivo que los atacantes tratan de comprometer. Los elementos son la confidencialidad, la integridad, la autenticación y la disponibilidad de recursos del sistema, bajo estos cuatro elementos el atacante tratará de explotar alguna de las vulnerabilidades para así poder encontrar una debilidad en alguno de estos elementos:

Tabla 2-2: Seguridad Informática

Indicador	Concepto
Confidencialidad	Un atacante puede estar atentando contra la confidencialidad de una persona de la organización al robar información importante como contraseñas u otros tipos de datos que viajan a través de las redes confiables y así poder tener acceso a información que solo esa persona la puede tener. Un ejemplo en el cual se encuentra un ataque de confidencialidad es el envenenamiento de la tabla ARP (Alvear, 2019).
Integridad	Un atacante podría interceptar un mensaje que se está transmitiendo a través de un protocolo de comunicación dentro de la red, en la cual el atacante cambia un bit del texto cifrado con el objetivo de alterar los datos del criptograma, este ataque se lo conoce comúnmente como BIT-FLIPPING y son considerados como ataque a la integridad de la información (Alvear, 2019).
Disponibilidad	Uno de los ataques que son más comunes y que se puede ver un ejemplo de ataque a la disponibilidad del servicio son DoS en el cual el atacante podría utilizar los recursos como el ancho de banda de la conexión del ISP para inyectar un número gigantesco de mensajes y forzar la caída de éste, negando así este servicio a los usuarios que realmente están utilizando el sistema (Alvear, 2019).
Autenticación	En la autenticación se debe saber que la persona con la que se están comunicando es la persona correcta, los ataques más conocidos implican engañar al sistema y se realiza tomando las sesiones ya establecidas por la víctima y así el atacante pueda obtener el usuario y contraseña de la víctima a la cual está suplantando (Alvear, 2019)

Realizado por: Centeno, Hernán, 2024

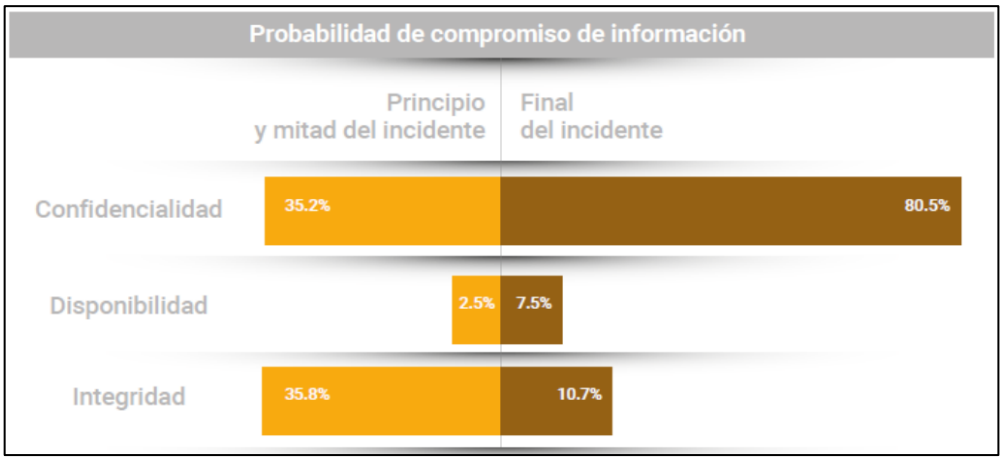


Figura 13-2: Probabilidad de compromiso de información

Fuente: (Fernandes et al., 2019)

2.2.6.1. Auditoria informática

La auditoría ofrece la habilidad de probar los sistemas, encontrar riesgo y comprobar si los controles son los apropiados para mitigar la exposición a los diferentes riesgos, cabe recalcar que la auditoría de seguridad no sólo trata de cómo ejecutar un sin número de herramientas de hackers, en un intento de entrar en la red. Su objetivo es evaluar la seguridad de la red interna de una empresa ante la posibilidad de recibir ataques por parte de un hacker que haya conseguido alcanzar la intranet o ataques provenientes del personal interno a la empresa (Miranda, 2019).

2.2.6.2. Tecnología defensiva en seguridad informática

Cuando se habla de tecnología defensiva en el ámbito de la informática, lo primero que se suele venir a la cabeza son los antivirus, pero hay mucha más tecnología que puede complementar la seguridad de lo organización. Lo primero que se debe tener en cuenta es que la responsabilidad de la gestión de la parte tecnológica de la defensa de la infraestructura, ha de recaer sobre el departamento de IT o del centro de operaciones de seguridad, es posible y cada vez más habitual que existan departamentos de seguridad específicos en muchas organizaciones y parte de su labor va a ser coordinado con el trabajo del Departamento de IT, por lo que hay que tener precaución en la gestión administrativa de esta situación (Romero et al., 2018).

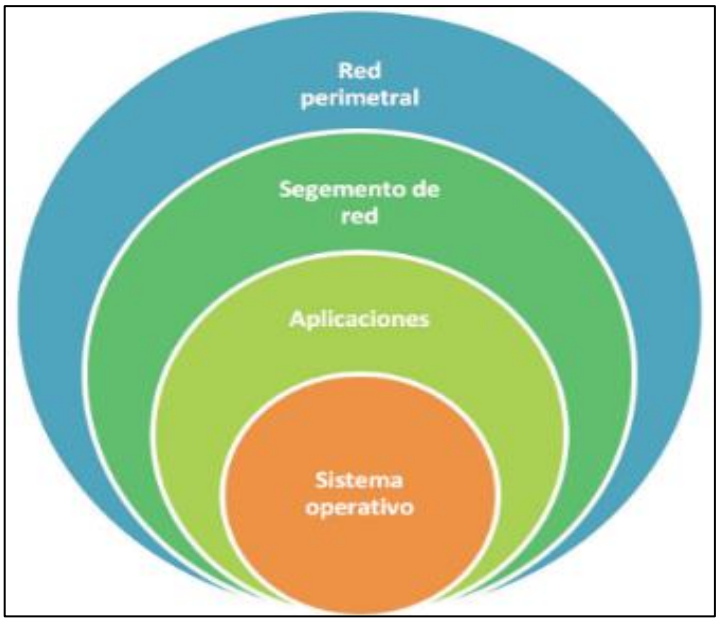


Figura 14-2: Tecnología defensiva en seguridad informática

Fuente: (Romero Castro et al., 2018)

2.3. Operacionalizacion de variables

Tabla 3-2: Operacionalización de variables

Variable	Tipo	Concepto
Módulo de seguridad	*Independiente*	El módulo de seguridad para un sistema web permite obtener un control de las distintas actividades que realiza un usuario dentro del sistema para proteger el acceso a los datos confidenciales (Soriano, 2014). Por lo tanto, la seguridad de sitios web eficaz requiere de esfuerzos de diseño a lo largo de la totalidad del sitio web (Fernández *et al.*, 2013).

Variable	Tipo	Concepto
Mejora la integridad de la información del sistema de E-salud de la ESPOCH	*Dependiente*	La seguridad no ha sido completamente integrada dentro del ciclo de vida de desarrollo y todavía es considerada después que el sistema ha sido diseñado (Rosado *et al.*, 2009). De esta manera, la seguridad Web se define como grado de protección de los datos, software y/o plataforma tecnológica de posibles pérdidas, actividades no permitidas o uso para propósitos no establecidos previamente (Huamaní, 2015). Para garantizar la integridad, de la información (Niño & Silega, 2018). La seguridad web a nivel de aplicación se refiere a mantener la información integra, es decir que no sea alterada bajo ningún concepto en aplicaciones web (independientemente de las tecnologías en las que se implemente o la seguridad del servidor web/base de datos back-end en la que se construye) (Scott & Sharp, 2002).

Realizado por: Centeno, Hernán, 2024

2.4. Matriz de consistenacia

Tabla 3-2: Matriz de concistencia

Variable	Indicadores	Dimensión	Técnica	Dimensión
Independiente: Módulo de seguridad	Metodologías existentes (ME) Lineamientos y reglas de seguridad (LR) Framework de JavaScript (FJ) Formatos de texto (FT)	OSSTMM	Revisión bibliográfica	– Módulos y estructura – Características – Fases de la metodología
		OWASP	Revisión bibliográfica	– Orientación de OWASP – Riesgos en seguridad de aplicaciones – OWASP top Aplicaciones Web – Protocolo HTTP – Servidor web – Arquitectura cliente servidor
		Angular JS	Observación directa- Escalas	– Node JS JavaScript del lado del Servidor – Servidor web web Node – Typescript – Características de Angular JS
		Servicios Json	Observación directa- Escalas	– Peticiones a un servicio JSON – Ventajas de los servicios
Dependiente: Mejora la integridad de la información del sistema de E-salud de la ESPOCH	Ataques de seguridad (AS) Evaluación del sistema (ES)	Ataques y seguridad	Observación directa- Escalas	– Fases de un ataque informático – Ethical hacking – Cyber ataques – Ransomware – Troyano – Inyecciones SQL – Principales problemas que causan los ataques SQL Injection – Acciones que se puede realizar mediante una inyección SQL
		Seguridad informática	Observación directa- Escalas	– Auditoria informática – Tecnología defensiva en seguridad informática

Realizado por: Centeno, Hernán, 2024

CAPÍTULO III

3. METODOLOGÍA DE INVESTIGACIÓN

En el presente capitulo se describe el proceso metodológico que fue empleado en la investigación, también se detalla los procedimientos, métodos y técnicas que permitieron recolectar información necesaria para obtener resultados que permitieron comprobar la hipótesis planteada para el presente trabajo realizado.

3.1. Tipo y diseño de la investigación

Para la siguiente investigación se define la metodología de estudio en este caso de definen los siguientes:

3.1.1. Estudios exploratorios

La investigación exploratoria tiene como objetivo examinar o explorar un problema de investigación poco estudiado o que no ha sido analizado antes. Por esa razón, ayuda a entender fenómenos científicamente desconocidos, poco estudiados o nuevos, apoyando en la identificación de conceptos o variables potenciales, identificando relaciones posibles entre ellas (Jose Luis, 2012).

3.1.2. Estudios descriptivos

La investigación descriptiva puede ser cuantitativa o cualitativa, incluyendo las colecciones de información cuantitativa que pueden ser tabuladas a lo largo de un continuo en forma numérica, como las puntuaciones en una prueba o el número de veces que una persona elija usar un cierto rasgo de un programa multimedia, o puede describir categorías de información como el género o los patrones de interacción cuando se utiliza la tecnología en una situación de grupo (Jose Luis, 2012).

3.1.3. Métodos, técnicas e instrumentos

El presente trabajo está basado en la aplicación de métodos, técnicas e instrumentos que permiten la interpretación y análisis de resultados obtenidos.

3.1.4. Métodos

Los métodos de investigación científica a utilizar siguen los siguientes pasos:

1. Método Científico.
2. Método hipotético deductivo.
3. Método experimental.
4. Método Sintético.

(Cienfuegos Velasco, 2019) ha determinado que el propósito principal de la ciencia (de la investigación científica y del método científico) es partir de las hipótesis y los objetivos (en este orden) para posteriormente establecer leyes y teorías (ciencia básica o pura). Sin embargo, en la práctica científica también se busca realizar investigación con leyes y teorías ya establecidas para intentar explicar hechos o fenómenos naturales y sociales.

Este método permite en la presente investigación permite la identificación del problema, además se realiza una revisión de conceptos relacionados con sistemas de seguridad web, brechas de seguridad, integridad de la información que alberga los sistemas para la web, entre otros conceptos de vital importancia para el desarrollo de este trabajo.

Las etapas establecidas para la aplicación de este método se describen a continuación:

1. Identificación y planteamiento del problema
2. Formulación de la hipótesis.
3. Recolección de información.
4. Análisis e interpretación de resultados.
5. Comprobación de la hipótesis.
6. Resultados obtenidos.

3.1.4.1. Método hipotético deductivo

A las hipótesis y al método hipotético deductivo se les acusa de otorgarle un aire de formalidad a la actividad científica. Más aun, suele afirmarse que las hipótesis no son necesarias para hacer Ecología aplicada. Aportamos argumentos para mostrar que esas afirmaciones no atienden adecuadamente el doble papel de las hipótesis en la investigación: proponer explicaciones a los patrones ecológicos, pero también guiar la toma de datos. Sugerimos que los escépticos con las hipótesis dudan, en realidad, de la capacidad de la Ecología de ofrecer hipótesis “explicativas” eficaces para formular predicciones. Esos ecólogos suelen tomar partido por el instrumentalismo

epistemológico pero usan hipótesis de manera implícita para guiar su investigación (Marone & Galetto, 2014).

Con el método hipotético deductivo en este estudio se observa y analiza la información confidencial que almacena el sistema médico de la ESPOCH, posteriormente se plantea la hipótesis de la investigación presente y por medio de la recolección de datos y durante el desarrollo del trabajo se determina la afirmación de la hipótesis planteada, para la utilización de este método se establece los siguientes pasos:

1. Observación del estado actual del sistema web.
2. Planteamiento de la hipótesis.
3. Comprobación de la hipótesis.

3.1.4.2. Método experimental

Mediante el uso de este método en la investigación se podrá identificar las variables de estudio que permiten realizar pruebas en base al ambiente de trabajo del sistema médico de la ESPOCH, (Murillo et al., 2012) afirman que en la investigación de enfoque experimental el investigador manipula una o más variables de estudio, para controlar el aumento o disminución de esas variables y su efecto en las conductas observadas. Dicho de otra forma, un experimento consiste en hacer un cambio en el valor de una variable (variable independiente) y observar su efecto en otra variable (variable dependiente). Esto se lleva a cabo en condiciones rigurosamente controladas, con el fin de describir de qué modo o por qué causa se produce una situación o acontecimiento particular.

3.1.4.3. Método sintético

Con el empleo del método sintético se plantea la identificación de alternativas de solución que permiten llegar a una conclusión como resultado de la investigación. (Jalal Caál et al., 2015) afirman que el Método Sintético es un proceso de razonamiento que tiende a reconstruir un todo, a partir de los elementos distinguidos por el análisis; se trata en consecuencia de hacer una explosión metódica y breve, en resumen.

En otras palabras, debemos decir que la síntesis es un procedimiento mental que tiene como meta la comprensión cabal de la esencia de lo que ya conocemos en todas sus partes y particularidades.

3.2. Técnicas

Durante el desarrollo de la investigación se identificaron y aplicaron las siguientes técnicas:

1. Experimentación.
2. Observación de campo.
3. Análisis de la información.

3.2.1. Instrumentos

Para el desarrollo del presente trabajo se hizo uso de varios instrumentos los mismo que permitieron la creación de un módulo de seguridad web el mismo que fue aplicado en el sistema médico de la ESPOCH.

3.2.1.1. Instrumentos software

Tabla 1-3: Instrumentos Software

NOMBRE	VERSIÓN	DESCRIPCIÓN	FUNCIONALIDAD
Visual Code	16.7	Editor de texto	Editor de texto utilizado para escribir el código para realizar el módulo de seguridad web.
SQL server	2016	Motor de base de datos	Motor de base de datos usado para el almacenamiento de la información que registra el módulo de seguridad web.
Angular	6	Framework de desarrollo	Framework en el cual se desarrolla el módulo de seguridad web.
Windows Server	2012	Servidor	Servidor en donde se va alojar el módulo de seguridad web, el mismo que debe ser configurado para que se ejecute la aplicación

Realizado por: Centeno, Hernán, 2024

3.2.1.2. Instrumentos hardware

Tabla 2-3: Instrumentos Hardware

NOMBRE	DESCRIPCIÓN	FUNCIONALIDAD
Computador	Procesador Core I7, memoria RAM de 16 GB, Disco duro de 1 TB	Computador desde el cual se codifica el módulo de seguridad web, luego de desarrollado y puesto a producción dicho módulo desde el mismo computador se puede dar mantenimiento.

Realizado por: Centeno, Hernán, 2024

3.2.1.3. Instrumentos bibliográficos

Tabla 3-3: Instrumentos bibliográficos

NOMBRE	DESCRIPCIÓN	FUNCIONALIDAD
Mendeley	Gestor de citas	Gestor usado para sistematizar las referencias adecuadamente dentro del documento, además es utilizado para almacenar los documentos utilizados en el desarrollo del documento.
Artículos científicos	Artículos publicados en revistas o congresos	Artículos en los cuales el o los autores presentan los resultados de una investigación, en este caso se toma como referencia artículos relacionado con las líneas de la investigación.
Tesis	Tesis de maestría relacionados con seguridad telemática	Documentos en los cuales se representa el estudio realizado en temas relacionados con seguridad telemática, sirve como guía para el desarrollo de este trabajo.
Sitios Web	Sitios web oficiales de las herramientas tecnológicas	Sitios web oficiales de las herramientas tecnológicas utilizadas para el desarrollo del módulo de seguridad web.

Realizado por: Centeno, Hernán, 2024

3.3. Procedimiento

Para el desarrollo de un módulo de seguridad web se identificaron una serie de pasos que se debe seguir los cuales establecen una guía para un desarrollo eficiente y que cumpla con las necesidades de tener un respaldo de seguridad de la información que alberga el sistema médico de la ESPOCH.

En el procedimiento a seguir se identificaron los siguientes pasos que se describe a continuación:

1. Análisis de la base de datos.
2. Identificación de información confidencial.
3. Verificar si existen copias de seguridad de la base de datos.
4. Usuarios que manipulan la información del sistema.
5. Creación de una tabla de auditoría del sistema.

5.1. Crear un registro donde se almacene el usuario que accede a la información en el sistema.
5.2. Crear un registro donde se almacene la fecha y hora que el usuario accede a la información del sistema.
5.3. Crear un registro donde se almacene la dirección de la computadora que el usuario accede a la información del sistema.
5.4. Crear un registro donde se almacene qué tipo de información tuvo acceso el usuario accede a la información del sistema.

6. Creación de backup de la base de datos periódicamente.
7. Certificados de seguridad SSL.
8. Control de ingreso de código.
   8.1. Control de las consultas que realiza el usuario en el sistema.
   8.2. Control de la información que ingresa el usuario en el sistema.
   8.3. Identificar si desde la interfaz el usuario ingresa código malicioso conocido como inyecciones SQL y bloquear el ingreso.

3.4. Planteamiento de la hipótesis

Con la implementación de un módulo de seguridad se mejora la integridad de la información del sistema de E-salud de la ESPOCH.

3.5. Operacionalización de las Variables

De la hipótesis planteada para la presente investigación se identificaron las siguientes variables:

3.5.1. Variable Independiente

Implementación de un módulo de seguridad.

3.5.2. Variable Dependiente

Mejora la integridad de la información del sistema de E-salud de la ESPOCH.

3.5.3. Operacionalización conceptual

En el desarrollo realizado de la siguiente investigación se identificaron las siguientes variables que de detallan a continuación:

Tabla 4-3: Operacionalización conceptual

VARIABLE	TIPO	CONCEPTO
Implementación de un módulo de seguridad	Independiente	En este caso la "Implementación de módulos de seguridad" es la variable independente debido que para realizar una evaluación solo se requiere de tener implementado el módulo de seguridad.
Mejora la integridad de la información del sistema de E-salud de la ESPOCH	Dependiente	En este caso "Mejora la integridad de la información del sistema de E-salud de la ESPOCH" es la variable dependiente debido que para asegurar la integridad de la información de un sistema se debe tener primero implementado un el módulo de seguridad.

Realizado por: Centeno, Hernán, 2024

3.5.4. Operacionalización metodológica

Tabla 5-3: Operacionalización Metodológica de Variables

VARIABLE	CATEGORÍA	INDICADOR	TÉCNICA	INSTRUMENTO
Implementación de un módulo de seguridad	Independiente	Seguridad Calidad	Desarrollo	Desarrollo de un módulo de seguridad de sistemas web.
Mejora la integridad de la información del sistema de E-salud de la ESPOCH	Dependiente	Integridad de la información Análisis Confidencialidad	Evaluación del módulo de seguridad	Implementación del módulo desarrollado y determinar si permite mejorar la integridad de la información del sistema médico de la ESPOCH

Realizado por: Centeno, Hernán, 2024

3.6. Ambiente de desarrollo y pruebas del módulo de seguridad

3.6.1. Ambiente de desarrollo

Para la elaboración de un módulo de seguridad de sistemas para la web en primera instancia se debe levantar un ambiente de desarrollo el mismo que permite la creación y realizar pruebas de funcionamiento de dicho módulo previo a su paso a producción en el sistema médico implementado en la ESPOCH.

Para el desarrollo de dicho módulo se identifica los siguientes módulos:

- **Identificación de información confidencial**
  Dentro del sistema médico se puede identificar información que es confidencial de los pacientes por lo tanto solo puede tener acceso el médico que atiende a dicho paciente, de igual manera existe información pública que puede ser vista por todo el personal de salud que labora en el centro médico.
- **Copias de seguridad de la base de datos**
  se procedió a analizar si se realiza copias de seguridad o backup de la base de datos del sistema médico, con el fin de facilitar este proceso y con el objetivo de tener copias de respaldo de la base de datos se opta automatizar dicho proceso.
- **Creación de una tabla de auditoria en la base de datos**
  Con el objetivo de tener un control acerca del acceso a la información confidencial del sistema médico se procede a crear una tabla dentro de la base de datos en la cual se registre el usuario que accede a la información de cada paciente, también se registra la fecha, hora, dirección ip del usuario y actividad realizada cuando accede a la información.
- **Análisis y prevención de código malicioso**
  Dentro del sistema médico los usuarios del mismo pueden realizar consultas por lo cual el sistema les permite ingresar los parámetros de la consulta a realizar, en estos parámetros el usuario puede ingresar código malicioso que al no ser controlado este código puede hacer caer los servicios del sistema o a su vez puede exponer información que no debería ser mostrada públicamente.

3.6.2. Ambiente de pruebas

Una vez desarrollado el módulo de seguridad bajo las primicias identificadas se procede con la siguiente etapa que es probar el funcionamiento de dicho módulo, para llevar a cabo las pruebas y comprobar que el módulo desarrollado cumpla con las expectativas establecidas se realiza los siguientes pasos en el ambiente de desarrollo o de pruebas:

- Acceso a la información confidencial del sistema.
- Modificación y eliminación de información confidencial.
- Pérdida de la base de datos.
- Eliminación de una o varias tablas de base de datos.
- Inserción de código malicioso.
- Navegar en sitios no seguros.

3.6.2.1. Escenario de prueba 1: Acceso a la información confidencial del sistema.

En este escenario se procede a acceder a la información de los pacientes desde los diferentes perfiles que dispone el sistema médico, entre la información que se accede está: Historias clínicas, reportes de atención médica entre otras, el objetivo de este escenario es guardar un registro del usuario que accede a la información y que tipo de información es la visualiza.

Tabla 6-3: Escenario de prueba 1: Acceso a la información confidencial del sistema

ESCENARIO	TIPO DE ATAQUE	CONTROL	CONSECUENCIA
Sin el módulo de seguridad web	Ver información de historias clínicas.	Ninguno	El usuario puede observar la información sin dejar rastro de la información que observa.
	Ver información de patologías.		
Sin el módulo de seguridad web	Ver información de citas médicas.	Módulo de seguridad	El módulo de seguridad crea un registro del usuario y perfil desde el cual se accede a la información.
	Ver información de consultas médicas		

Realizado por: Centeno, Hernán, 2024

3.6.2.2. Escenario de prueba 2: Modificación y eliminación de información confidencial.

En este escenario se procede a modificar información de los pacientes como son Historias clínicas y también se procede a la eliminación de datos como son citas médicas, el objetivo de este escenario es guardar un registro del usuario que accede a la información y que tipo de información es la visualiza.

Tabla 7-3: Escenario de prueba 2: Modificación y eliminación de información confidencial

ESCENARIO	TIPO DE ATAQUE	CONTROL	CONSECUENCIA
Sin el módulo de seguridad web	Modificar o eliminar información de historias clínicas.	Ninguno	El usuario puede modificar o eliminar la información sin dejar rastro de la información que observa.
	Modificar o eliminar información de patologías.		
Sin el módulo de seguridad web	Modificar o eliminar información de citas médicas.	Módulo de seguridad	El módulo de seguridad crea un registro del usuario y perfil desde el cual se elimina o modifica la información.
	Modificar o eliminar información de consultas médicas		

Realizado por: Centeno, Hernán, 2024

3.6.2.3. Escenario de prueba 3: Pérdida de la base de datos o eliminación de una o varias tablas de base de datos.

En este escenario se procede a borrar gran parte de la información que alberga la base de datos o a su vez también se procede a la eliminación de datos históricos de los pacientes registrados en el sistema médico, el objetivo de este escenario es poder restaurar la información eliminada o perdida mediante copias de seguridad de la base de datos.

Otra amenaza identificada es la eliminación de ciertas tablas de la base de datos lo que impide que el sistema funcione y también se procede a la eliminación de la base de datos por completo, el objetivo de este escenario es poder restaurar la base de datos mediante copias de seguridad generadas automáticamente de la base de datos.

Tabla 8-3: Escenario de prueba 3: Pérdida de la base de datos

ESCENARIO	TIPO DE ATAQUE	CONTROL	CONSECUENCIA
Sin el módulo de seguridad web	Borrado accidental de la toda la base de datos.	Ninguno	Pérdida de toda la información albergaba en la base datos desde sus inicios hasta la fecha de pérdida de información.
	Borrado accidental de toda la información contenida en la base de datos.	Ninguno	
	Eliminación de una o varias tablas de la base de datos.	Ninguno	
Sin el módulo de seguridad web	Borrado accidental de la toda la base de datos.	Módulo de seguridad	Restauración de la última copia de seguridad generada automáticamente en el servidor de bases de datos.
	Borrado accidental de toda la información contenida en la base de datos.	Módulo de seguridad	
	Eliminación de una o varias tablas de la base de datos.	Módulo de seguridad	Proceso de restauración de las tablas eliminadas para su posterior migración de datos de la última copia de respaldo.
	Borrado de información de una o varias tablas de la base de datos.	Módulo de seguridad	Proceso migración de datos de la última copia de respaldo.

Realizado por: Centeno, Hernán, 2024

3.6.2.4. Escenario de prueba 4: Inserción de código malicioso.

En este escenario se procede a insertar código malicioso en las opciones de consulta que presenta el sistema, la intención es colapsar los servicios y provocar que se caigan los mismo y como

consecuencia queden expuestas las direcciones de los servicios, también se procede a la inserción de otro tipo de código conocido como de sentencias SQL , el objetivo este escenario es tener un control del código que ingresa el usuario y poder determinar si es código malicioso el módulo de seguridad bloquee el paso del mismo.

Tabla 9-3: Escenario de prueba 4: Inserción de código malicioso

ESCENARIO	TIPO DE ATAQUE	CONTROL	CONSECUENCIA
Sin el módulo de seguridad web	Ingreso de signos especiales.	Ninguno	Provoca confundir a los servicios de tal manera que al no reconocer estos signos el servicio colapsa dejando expuesta la ruta del mismo en la consola del navegador.
	Ingreso de sentencias maliciosas.	Ninguno	Logra confundir a los servicios de tal manera que exponen más información de la que debería mostrar
Sin el módulo de seguridad web	Ingreso de signos especiales.	Módulo de seguridad	El módulo de seguridad controla el texto que ingresa el usuario desde la interfaz y en caso de ser sentencias sospechosas bloquea el paso hacia los servicios.
	Ingreso de sentencias maliciosas.	Módulo de seguridad	

Realizado por: Centeno, Hernán, 2024

3.6.2.5. Escenario de prueba 5: Navegar en sitios no seguros

En este escenario la intensión desplegar el sistema médico de la ESPOCH como sitio no seguro es decir que la navegación sea por HTTP, de tal manera que el navegador web reconozca la dirección sistema como sospechosa.

Tabla 10-3: Escenario de prueba 5: Navegar en sitios no seguros

ESCENARIO	TIPO DE ATAQUE	CONTROL	CONSECUENCIA
Sin el módulo de seguridad web	Desplegar el sistema en sin certificados de seguridad	Ninguno	El sistema por los navegadores web será reconocido como sitio no seguro.
Sin el módulo de seguridad web	Desplegar el sistema en con certificados de seguridad	Instalación de certificados SSL	Al instalar certificados SSL el sitio será reconocido como sitio seguro, de tal manera que el sistema no será susceptible a la exposición de información como variables de sesión.

Realizado por: Centeno, Hernán, 2024

3.6.2.6. Entorno gráfico del escenario de prueba sin el módulo de seguridad web

A continuación, se presenta una figura en la que se describe el escenario de prueba sin la implementación del módulo de seguridad.

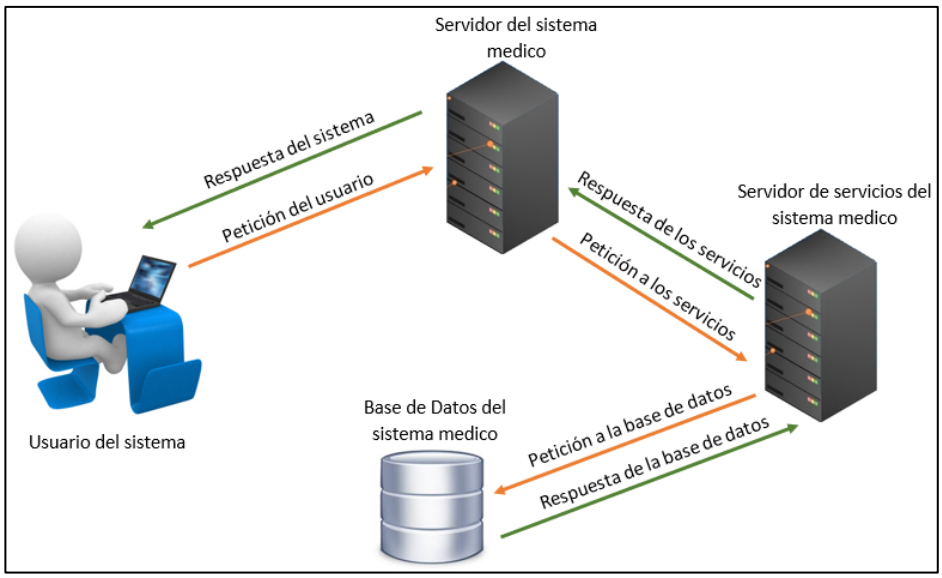


Figura 1-3: Escenario de prueba sin el módulo de seguridad web

Realizado por: Centeno, Hernán, 2024

3.6.2.7. Entorno gráfico del escenario de prueba con el módulo de seguridad web

A continuación, se presenta una figura en la que se describe el escenario de prueba luego de la implementación del módulo de seguridad.

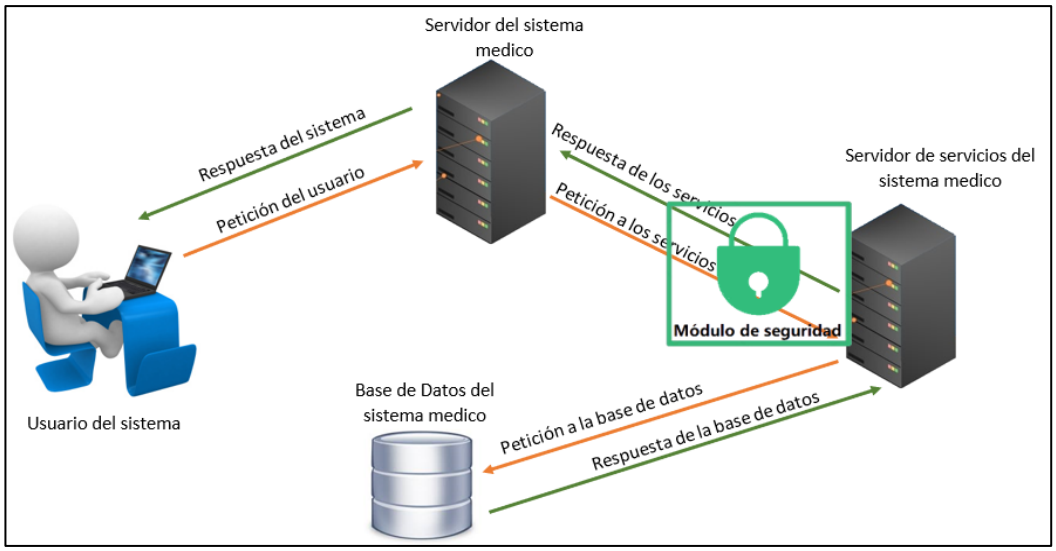


Figura 2-3: Escenario de prueba implementado el módulo de seguridad web.

Realizado por: Centeno, Hernán, 2024

3.6.3. Objetivos del módulo de seguridad para sistemas en la web

Tomando como referencia los lineamientos del OWASP TOP 10 se establecen los objetivos que debe cumplir el módulo de seguridad para sistemas web, también se implementa un objetivo en base a las líneas de OSSTMM que trabaja de manera sincronizadas con los demás objetivos logrando de esta manera elaborar un módulo de seguridad confiable para los usuarios y administradores del sistema médico de la ESPOCH. En la tabla 11-3 se detalla los objetivos y la metodología a la que corresponde cada uno:

Tabla 11-3: Objetivos del módulo de seguridad para sistemas en la web

NO.	OBJETIVO	CORRESPONSAL
1	Tener un control de los usuarios que acceden a la información sensible que alberga el sistema.	OWASP - A3: Exposición de datos sensibles
2	Llevar un control de los usuarios que pueden modificar o a su vez es eliminar información sensible del sistema.	OWASP - A5: Pérdida de Control de Acceso
3	Automatizar el proceso que permita generar periódicamente copias de respaldo de la base de datos del sistema.	OWASP - A3: Exposición de datos sensibles
4	Implementar un control entre la capa de la interfaz del sistema y los servicios que permita detectar el ingreso de código malicioso.	OWASP - A1: Inyección SQL
5	Implementación de certificados SSL en el sistema médico de la ESPOCH	OWASP - A2: Pérdida de Autenticación
6	Implementación de una tabla de auditoría de las actividades realizadas en el sistema	OSSTMM

Realizado por: Centeno, Hernán, 2024

3.6.3.1. Tener un control de los usuarios que acceden a la información sensible que alberga el sistema

El objetivo es de tener una tabla de auditoría que registre los movimientos de todos los usuarios del sistema que tienen acceso a información sensible, la intensión es identificar lo siguiente:

- Fecha y hora de acceso.
- Ubicación desde donde accede al sistema.
- Perfil de usuario que accede a la información.
- Acción que realiza sobre la información.
- Coordenadas de donde se encuentra el computador desde el que acceden al sistema.
- Dirección IP del computador desde el cual accede a la información.

3.6.3.2. Llevar un control de los usuarios que pueden modificar o a su vez es eliminar información sensible del sistema.

El objetivo es conceder permisos de edición y eliminación de información del sistema médico solo a los profesionales de la salud que realizan consultas médicas debido a que este tipo de

información debe ser confidencial entre el paciente y el profesional de la salud. En este caso también se lleva un registro de los movimientos que hace cada profesional identificando el usuario y perfil desde donde está siendo manipulada la información.

3.6.3.3. Automatizar el proceso que permita generar periódicamente copias de respaldo de la base de datos del sistema.

El objetivo es tener un proceso automatizado que genere periódicamente backup de la base de datos del sistema médico, en este caso se establece que se debe realizar dos copias de seguridad en el día, la intensión es evitar la pérdida de información en el sistema ya sea que suceda por factores internos o externos a la institución.

3.6.3.4. Implementar un control entre la capa de la interfaz del sistema y los servicios que permita detectar el ingreso de código malicioso.

El objetivo es tener un control intermedio entre la vista y los servicios JSON del sistema, hay que considerar que el sistema permite realizar consultas a los usuario para los cual se debe ingresar los criterios de búsqueda, en estos criterios que ingresa el usuario puede contener código malicioso y la intensión de detectar la inserción de dicho código ya que puede exponer información del sistema o a su vez colapse los servicios, la idea fundamental es bloquear el paso de este código hacia el consumo de los servicios y permitir el paso de código considerado como no malicioso de esta manera el módulo de seguridad sirve como escudo protector del sistema médico.

3.6.3.5. Implementación de certificados SSL en el sistema médico de la ESPOCH

El objetivo es implementar en el despliegue del sistema certificados SSL con la intensión que el sistema sea reconocido como seguro, de esta manera se evita que la información que viaja hacia el servidor sea susceptible de robo, además en este objetivo se establece que al cerrar la sesión de usuario dentro de la codificación se destruya por completo todas las variables se sesión asegurando de cerrar por completo el sistema.

3.6.3.6. Implementación de una tabla de auditoría de las actividades realizadas en el sistema

El objetivo es implementar en la base de datos una tabla que lleve un registro de las actividades que realiza cada usuario en el sistema, la intensión es estar preparados para casos extremos como

divulgación de información confidencial del sistema médico entre otros casos que se pueden presentar.

3.6.4. Componentes del módulo de seguridad web

El módulo de seguridad está constituido por cuatro componentes fundamentales con las cuales se pretende asegurar la integridad de la información del sistema médico, en la tabla 12-3 se describe de manera detallada dichos componentes.

Tabla 12-3: Componentes del módulo de seguridad web

NO.	COMPONENTE	FUNCIÓN
1	Registro de actividades dentro del sistema (Auditoría)	Tener un registro de las actividades que realiza sobre el sistema. (Yeiniel et al., 2012) considera que una auditoria informática es un conjunto de procedimientos y técnicas que permiten evaluar, total o parcialmente, el grado en que se cumplen la observancia de los controles internos asociados al sistema informático
2	Prevención de Inyecciones SQL	Tener un control que permita determinar y bloquear el paso de texto que sea considerado como código malicioso. (Chalabe Jimenez, 2019)considera que las inyecciones SQL es actualmente uno de los ataques web más comunes e importantes.
3	Certificados SSL	Algunos protocolos pueden ser simple HTTP lo cual no es seguro, los certificados SSL permite transferencia segura de información a través de Internet por HTTPS (Romero Castro et al., 2018), con el fin de tener conexiones seguras en el sistema médico se hace la instalación de certificados SSL y de esta manera el sistema será reconocido como sitio seguro.
4	Copias de seguridad de la base de datos	Todo sistema está expuesto a que su información se pierda o se borre ya sea por factores internos o externos por esta razón se automatiza el proceso de creación de copias de respaldo de la base datos como un plan de contingencia ante la pérdida de información. (Peña, 2016) define un backup como una copia adicional de la información que puede utilizarse con fines de recuperación y restauración ante fallos.

Realizado por: Centeno, Hernán, 2024

Los módulos definidos en la tabla 12-3 son los pilares fundamentales que constituyen el módulo de seguridad desarrollado, la intensión de este módulo es tener un registro de las actividades que realizan los usuarios en el sistema, prevenir la inserción de código malicioso que atente contra la información del sistema, así como también resguardar la información en caso de pérdida o fallos en la base de datos.

Los componentes del módulo de seguridad se dividen o se clasifica en dos grupos según la funcionalidad y forma de desarrollo de cada uno, en la tabla 13-3 se detalla la clasificación de los componentes del módulo de seguridad.

Tabla 13-3: Clasificación de los componentes del módulo de seguridad web

GRUPO	COMPONENTE	CRITERIO
Grupo 1	Registro de actividades dentro del sistema	Estos componentes corresponden a desarrollo de software, ya que para el registro de las actividades en el sistema dentro de la base de datos se creó una tabla de auditoría en la que se almacena esta información, en lo que concierne a prevención de inyecciones SQL dentro de la programación se incluye un control de los parámetros que ingresa el usuario al sistema.
	Prevención de Inyecciones SQL	
Grupo 2	Certificados SSL	Estos componentes se clasifican en un segundo grupo debido que para su implementación no se requiere de actividades de desarrollo de software, estos componentes están orientados directamente al servidor donde se aloja las aplicaciones.
	Copias de seguridad de la base de datos	

Realizado por: Centeno, Hernán, 2024

3.6.4.1. Acoplamiento del módulo de seguridad en sistemas para la web

La principal idea de este módulo de seguridad desarrollado es que se pueda acoplar en cualquier sistema para la web.

Debido a la estructura que presentan los componentes que hacen referencia a desarrollo de software no pueden ser adaptados en cualquier tipo de estructura de sistemas web por lo que se requiere que el sistema web en el que se va a implementar estos componentes debe mantener una estructura cliente servidor es decir que tenga definido su capa de servicios separada de la capa de interfaz de usuario ya que los componentes de desarrollo intervienen como una capa intermedia entre la vista y servicios en la figura 3-3 se muestra la aplicación de los componentes de desarrollo.

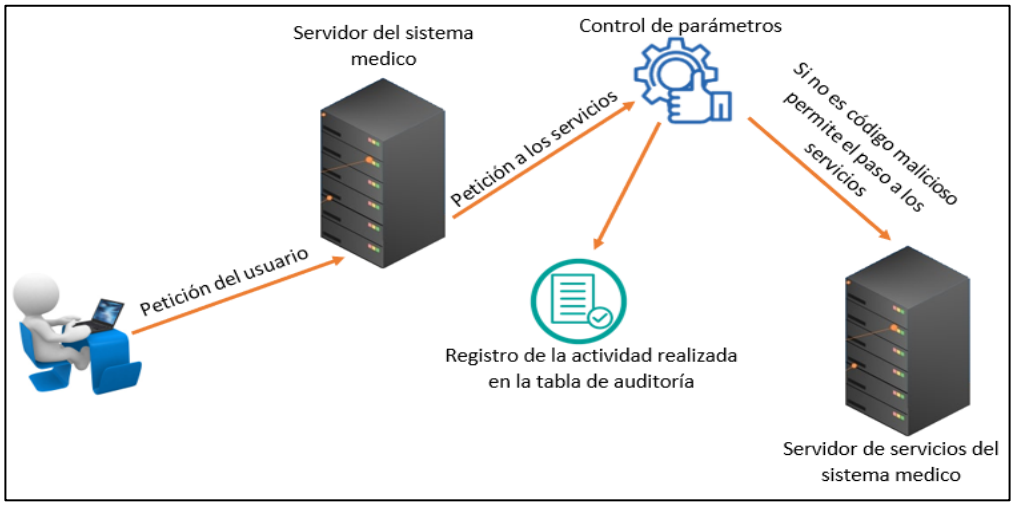


Figura 3-3: Componentes que corresponden a desarrollo de software

Realizado por: Centeno, Hernán, 2024

Estos módulos funcionan de manera sincronizada, en primera instancia el módulo de control de datos verifica que el usuario no esté intentando acceder código malicioso, posterior a este control el siguiente módulo procede a crear un registro de la actividad que realiza el usuario dentro del sistema. En la figura 4-3 de muestra el flujo de trabajo de estos módulos.

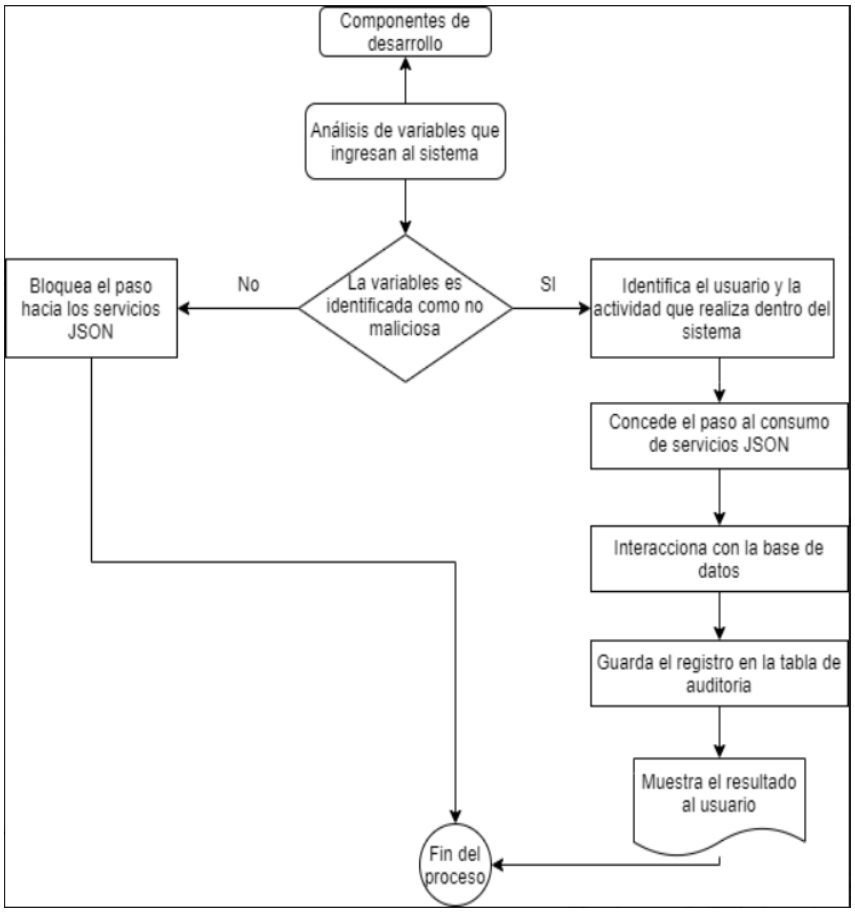


Figura 4-3: Flujo de trabajo

Realizado por: Centeno, Hernán, 2024

Los componentes que se ejecutan en el servidor pueden ser adaptados a cualquier tipo de sistema para web sin importar la estructura que mantenga dentro de su codificación, esto se debe que en el servidor se debe instalar certificados SSL que permita conexiones seguras y de igual manera para los respaldos de la base de datos se debe configurar un script que permita realizar esta actividad.

En la figura 5-3 se puede visualizar el diagrama de flujo de trabajo del proceso de copias de respaldo de la base de datos en el servidor.

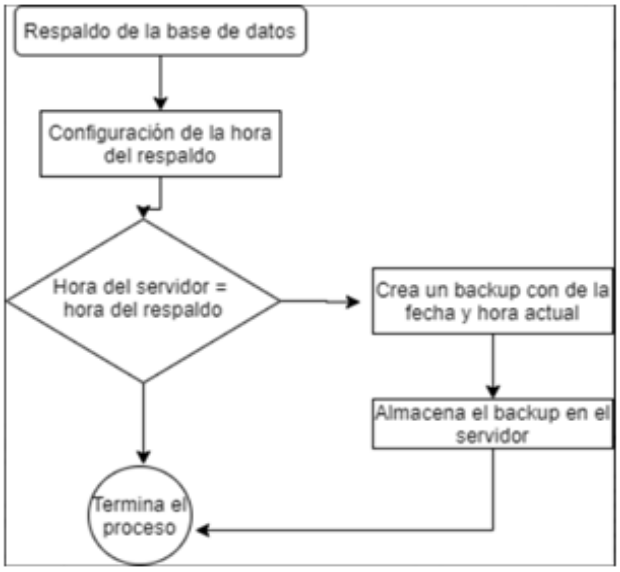


Figura 5-3: Flujo de trabajo copias de seguridad

Realizado por: Centeno, Hernán, 2024

CAPÍTULO IV

4. RESULTADOS Y DISCUSIÓN

La información es el recurso más valioso e importante que dispone toda organización, empresa o entidades públicas, con el propósito de tener la información almacenada digitalmente se procede a la creación de sistemas informáticos los mismos que en conjunto con las bases de datos gestionan y almacenan la información, con el hecho de tener sistemas y procesos automatizados son susceptibles a ataques informáticos que atentan con la seguridad de la información.

Bajo esta premisa el estudio de la presente investigación se basa en la implementación de un módulo de seguridad que permita elevar el nivel de seguridad en el sistema médico de la ESPOCH para lo cual el módulo fue implementado en el sistema que se encuentra en producción para realizar las pruebas pertinentes y los resultados obtenidos se describen en el presente capítulo.

4.1. Implementación del módulo de seguridad en el sistema médico de la ESPOCH

Para la validación del módulo de seguridad propuesto se procede a la implementación dentro del sistema médico de la ESPOCH, dicho módulo está constituido por cuatro componentes los mismo que a la vez se clasifican en dos grupos de acuerdo con la funcionalidad que cumplen y manera de ejecución de dentro del sistema médico.

El primer grupo corresponde a la interacción directa con el sistema médico y está conformado por dos módulos que son:

1. Análisis de los datos que ingresa el usuario.
2. Registro de la actividad que realiza el usuario.

El segundo grupo corresponde a la interacción con los servidores de producción que se utilizó para el despliegue del sistema médico que son el servidor de la vista y el servidor de servicios. Los módulos que componen este grupo son:

1. Certificados SSL.
2. Copias de seguridad de la base de datos.

4.1.1. Análisis de los datos que ingresa el usuario

El primer filtro que realiza el módulo de seguridad desarrollado es el control de los datos que ingresa el usuario en el sistema, este componente se aplica como un ente intermedio en la capa de la vista y los servicios que consume el sistema. La funcionalidad es analizar los datos que ingresa en el sistema y si el componente lo detecta como código malicioso bloquea el paso a los servicios y muestra un mensaje de error al usuario.

Para este ejemplo se toma como referencia la búsqueda de historias clínicas para lo cual el usuario tiene varios criterios de búsqueda que son:

1. Buscar por número de cedula.
2. Buscar por nombres.
3. Buscar por apellidos.

Si el sistema identifica que cualquiera de estos tres criterios de búsqueda contiene código malicioso muestra un mensaje de error y no permite el paso al consumo de servicios.

En la figura 1-4 se muestra el funcionamiento del sistema con implementado el componente de control de datos para lo cual en el campo de criterio de búsqueda se colocó el parámetro siguiente: apostrofe – asterisco – apostrofe (‘*’) este dato ingresado es considerado como código malicioso debido que en una sentencia SQL el asterisco implica que devuelva toda la información en vez de un limitado campos que el usuario debería visualizar.

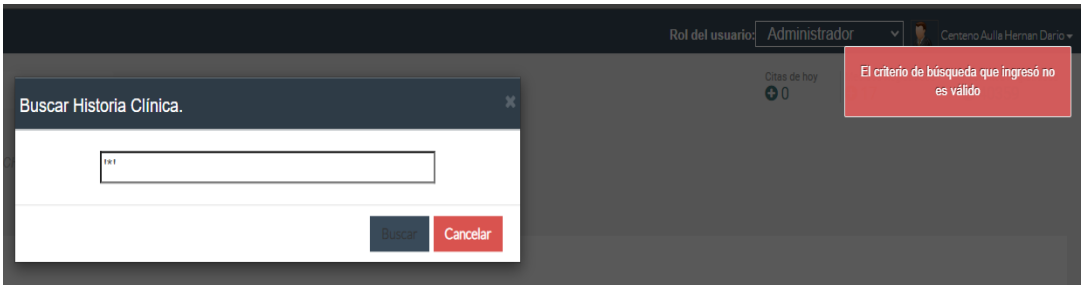


Figura 1-4: Control de datos implementado el módulo de seguridad

Realizado por: Centeno, Hernán, 2024

Para visualizar el comportamiento del sistema sin la implementación del componente de control de código se procede a ingresar el mismo parámetro es: apostrofe – asterisco – apostrofe (‘*’), en bases datos como MySql este tipo de parámetros da resultados ya que devuelve toda la información que contiene la o las tablas sobre la cual se está se realizando la consulta, en motores

de bases de datos como SQL que es el caso del sistema médico el gestor de base de datos identifica como una sentencia no válida y genera un error en la sintaxis de la sentencia y como consecuencia devuelve el error en la consola de los navegadores en la cual notifica que el servicio no fue encontrado y muestra la URL del servicio que se quiere hacer uso, en la figura 2-4 se puede ver el resultado de realizar este tipo de consultas sin el módulo de seguridad planteado.

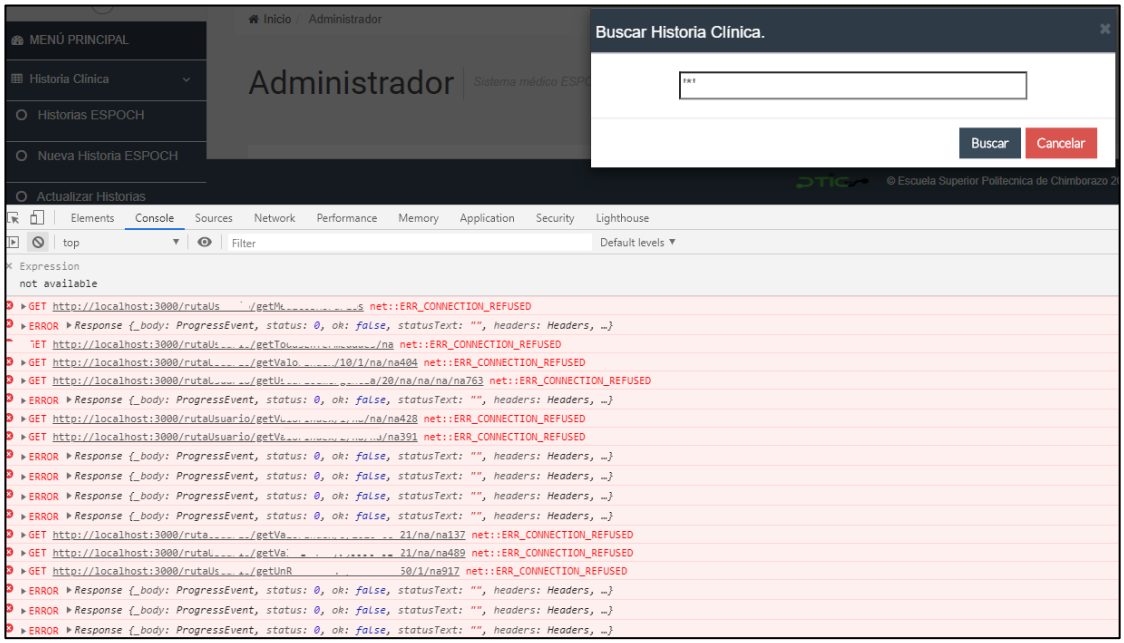


Figura 2-4: Inserción de datos sin control

Realizado por: Centeno, Hernán, 2024

Esta prueba fue realizada en localhost y como se puede apreciar sin el componente de control al insertar código identificado como malicioso los servicios al no poder realizar la petición devuelve un error en el cual deja expuesto lo siguiente:

1. Ruta del servicio
2. Puerto por el que escucha el servicio
3. Método que consulta el servicio.
4. Numero de parámetros que debería recibir el servicio

4.1.2. Registro de la actividad que realiza el usuario

Este componente se ejecuta después que el componente de control de datos permite el paso a la consulta de los servicios del sistema médico, la funcionalidad de este componente es crear un registro de todas las actividades que se realiza dentro del sistema y dicha información la almacena en una tabla de auditoria, este componente es importante ya que se pueden presentar varios casos

de que atenten con la integridad del sistema un caso por ejemplo puede ser la divulgación de información confidencial, en este escenario se puede imprimir en reporte de la persona que tuvo acceso a esa información, los parámetros que se guardan en esta tabla son:

1. Cedula del usuario que ingresa.
2. Rol de usuario desde el cual ingresa.
3. Fecha y hora de ingreso.
4. Actividad que realiza en el sistema (Ver, insertar, modificar o eliminar información).
5. Datos que ingresa al sistema para realizar la actividad que desea.
6. Dirección IP de la computadora desde la cual ingresa al sistema.
7. Ciudad desde la cual ingresa.

En la figura 3-4 se muestra el resultado de la tabla de auditoria y de la información que alberga según las actividades que realizan dentro del sistema médico.

idAuditoria	perfil	usua...	metodo	accion	opcion	parametros	direccion	ciudad	fecha
159	MEDICO	06042...	ingresarEmer...	INGRESAR / EDIT...	27	Estado Eva: 1 - Detalle Eva...	45.184.1...	Riobamba	2020-08-14 14:59:00....
160	MEDICO	06042...	ingresarEmer...	INGRESAR / EDIT...	27	Estado Eva: 1 - Detalle Eva...	45.184.1...	Riobamba	2020-08-14 15:00:00....
161	MEDICO	06042...	ingresarEmer...	INGRESAR / EDIT...	27	Estado Eva: 1 - Detalle Eva...	45.184.1...	Riobamba	2020-08-14 15:00:00....
162	MEDICO	06042...	ingresarReser...	INGRESO TURNO ...	0	HCE: 0603356080101 - Fec...	45.184.1...	Riobamba	2020-08-14 15:03:00....
163	PACIENTE	06053...	ingresoReser...	RESERVA CITA ME...	0	HCE: null - Fecha_ 2020-8...	190.152....	Riobamba	2020-08-14 16:24:00....
171	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 0250157120101 - Fec...	2800:370...	Riobamba	2020-08-14 17:39:00....
172	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 0604837427101 - Fec...	2800:370...	Riobamba	2020-08-14 17:43:00....
173	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 0603988015101 - Fec...	2800:370...	Riobamba	2020-08-14 17:44:00....
174	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 1650093733101 - Fec...	2800:370...	Riobamba	2020-08-14 17:45:00....
175	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 0605524313101 - Fec...	2800:370...	Riobamba	2020-08-14 17:47:00....
176	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 2100760962101 - Fec...	2800:370...	Riobamba	2020-08-14 17:48:00....
177	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 0603461765101 - Fec...	2800:370...	Riobamba	2020-08-14 17:51:00....
178	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 0604454728101 - Fec...	2800:370...	Riobamba	2020-08-14 17:53:00....
179	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 0604221317101 - Fec...	2800:370...	Riobamba	2020-08-14 17:54:00....
180	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 0605653724101 - Fec...	2800:370...	Riobamba	2020-08-14 17:55:00....
181	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 0604868174101 - Fec...	2800:370...	Riobamba	2020-08-14 17:57:00....
182	ODONTOLOGO	06018...	ingresarReser...	INGRESO TURNO ...	0	HCE: 0605481134101 - Fec...	2800:370...	Riobamba	2020-08-14 17:58:00....

Figura 3-4: Registro de las actividades del sistema en la tabla de auditoría

Realizado por: Centeno, Hernán, 2024

4.1.3. Certificados SSL

Este componente se ejecuta o se despliega en los servidores de producción tanto de la vista como de los servicios del sistema médico que es en donde se instala los certificados SSL con el fin que la transferencia de datos sea segura al igual que el sistema desde los navegadores sea reconocido como sitio seguro.

El primer paso es colocar los certificados dentro del servidor como se puede apreciar en la figura 4-4.

Figura 4-4: Certificados SSL dentro del servidor

Realizado por: Centeno, Hernán, 2024

En segundo lugar, se debe colocar los servicios dentro del sitio web creado para el sistema que se quiere agregar los certificados SSL como se puede apreciar en la figura 5-4.

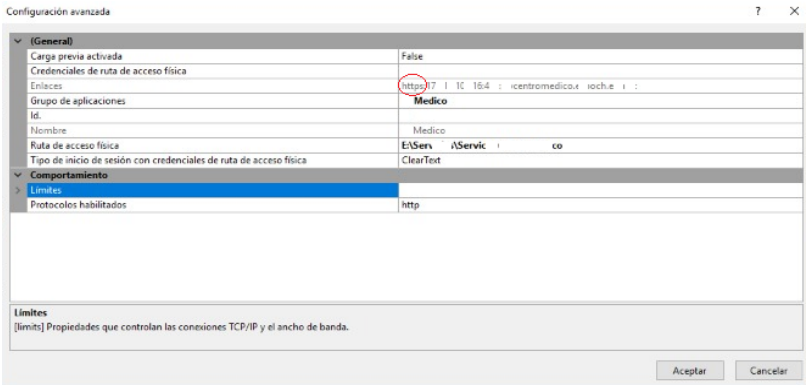


Figura 5-4: Agregar certificados SSL al sitio web

Realizado por: Centeno, Hernán, 2024

Finalmente, cuando el usuario va a navegar en el sistema el navegador reconoce el certificado y la URL del sistema pasa de HTTP a HTTPS es decir es reconocida como un sitio seguro, en la figura 6-4 se puede visualizar el funcionamiento de los certificados en el sistema médico.

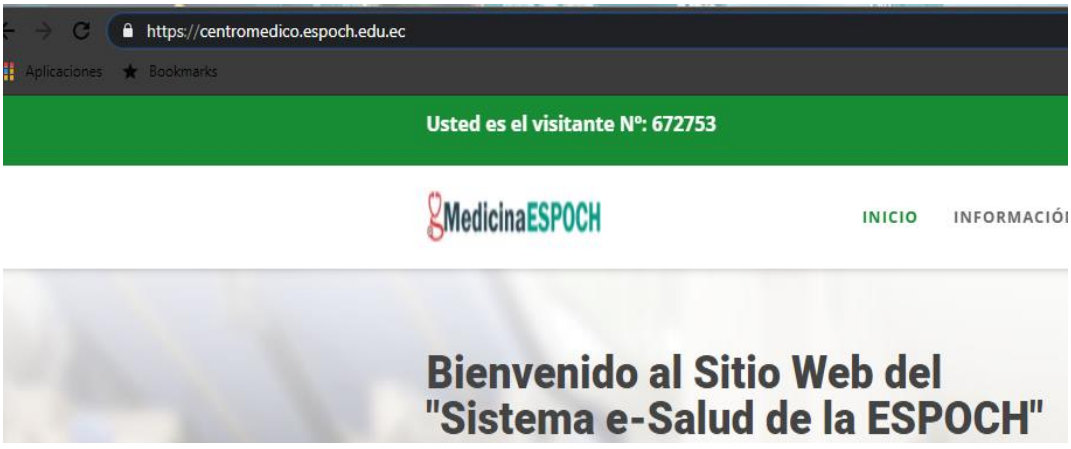


Figura 6-4: Sitio con el protocolo HTTPS

Realizado por: Centeno, Hernán, 2024

También dentro del mismo navegador se pruebe observar información más detallada del certificado que tiene el sitio implementado, entre la información que se puede observar es: la entidad certificadora, fecha de valides y caducidad del certificado entre otros datos, como se puede observar en la figura 7-4.

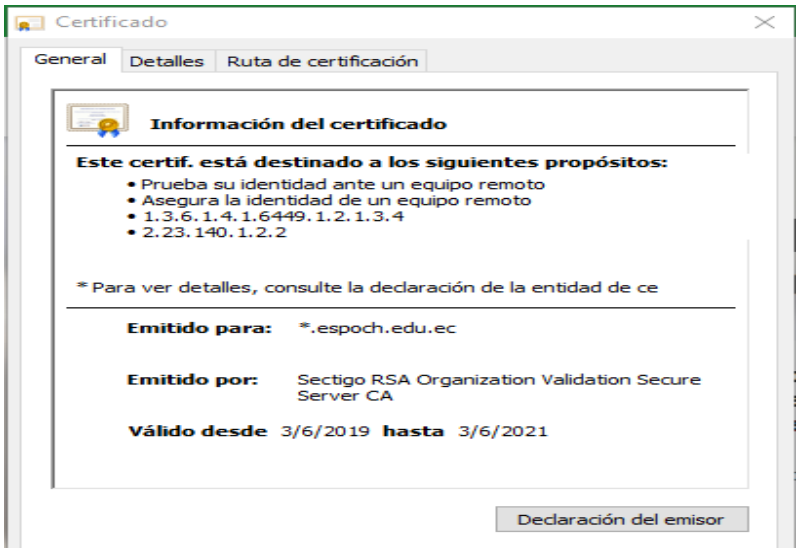


Figura 7-4: Información del certificado SSL del sitio web

Realizado por: Centeno, Hernán, 2024

4.1.4. Copias de seguridad de la base de datos

En este módulo se crea un proceso automatizado el mismo que se debe ejecutar en el servidor de base de datos con la finalidad que realice periódicamente copias de seguridad o backup de la base de datos, para este componente está establecido que las copias de seguridad deben ser dos por día a las 12 horas y a las 24 horas, en la figura 8-4 se puede observar el historial de las copias de seguridad que se genera.

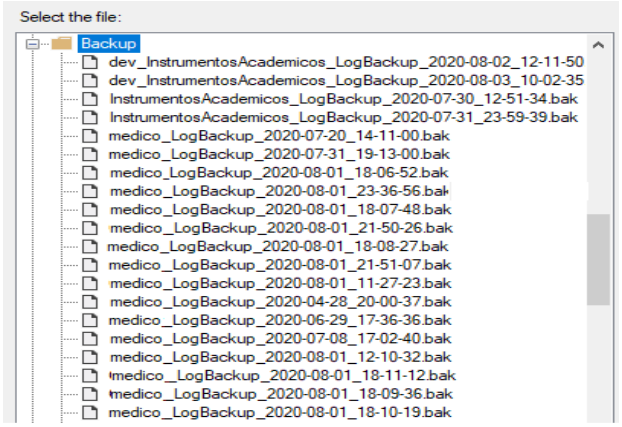


Figura 8-4: Historial de backups de la base de datos

Realizado por: Centeno, Hernán, 2024

4.2. Análisis de la contribución del módulo de seguridad web

En el presente proyecto se identificaron 4 componentes a partir de los cuales se identificaron 4 variables que permitieron medir de manera cuantitativa la aportación del módulo desarrollado dentro del sistema médico de la ESPOCH. Los componentes identificados son los siguientes:

1. Análisis de los datos que ingresa el usuario.
2. Registro de la actividad que realiza el usuario.
3. Certificados SSL
4. Copias de seguridad de la base de datos

Los componentes identificados reflejan el nivel de seguridad de seguridad e integridad de la información que presenta el sistema médico antes y después de implementar el módulo de seguridad desarrollado, al realizar esta comparación se obtiene como resultado un valor o un criterio, que tiene como finalidad determinar si la integración del módulo de seguridad aporta o no en la seguridad de la información del sistema médico.

Para el procesamiento y evaluación de la información se identificará las siguientes abreviaturas:

- **Número:** Numeración de los componentes.
- **Componente:** Nombre del componente desarrollado.
- **Control:** Tipo de control que estable con la integración del módulo de seguridad.
- **Actividades:** Las actividades que realiza el componente identificado.

Tabla 1-4: Componentes control y actividades desarrolladas

No.	Componente	Control	Actividades
1	Análisis de los datos que ingresa el usuario	Análisis de los datos que ingresa el usuario.	Realizar un control a nivel de interfaz de los datos que ingresa el usuario en el sistema.
		Permitir el paso al consumo de servicios.	Si los datos no son considerados como malicioso permitir el consumo de los servicios.
		Bloquear el paso al consumo de servicios.	Si los datos ingresados son considerados como malicioso mostrar un mensaje de error.
2	Registro de la actividad que realiza el usuario	Identificar el usuario que ingresa al sistema.	Identificar el nombre de usuario
			Identificar el rol del usuario.
		Identificar la activad a realizar en el sistema.	Ingreso de información
			Editar información
			Ver información
			Eliminar información
		Identificar la fecha y hora de acceso.	Calcular la fecha y hora del computador desde que ingresa el usuario al sistema.
		Identificar información de acceso.	Dirección IP del computador desde que ingresa al sistema.
			Lugar desde donde accede el usuario al sistema.
3	Certificados SSL	Instalación de certificados SSL.	Tener un protocolo de navegación HTTPS.

4	Copias de seguridad de la base de datos	Realizar copias respaldo de la base de datos.	Generar un script que se ejecute periódicamente
			Configurar que el script se ejecuta dos veces al día

Realizado por: Centeno, Hernán, 2024

4.3. Valoración de los índices de los componentes

Para la valoración de los índices se asignaron valores a cada uno de los componentes antes y después de la integración del módulo de seguridad, estos valores fueron asignados según el comportamiento de la información que tiene ante varias situaciones que se puede presentar en el uso sistema y que puede afectar con la integridad y seguridad de la información.

4.3.1. Criterios de evaluación

En este apartado se describen los valores cualitativos y cuantitativos referente al nivel de seguridad que se establecen a los parámetros identificados para la evaluación del grado de seguridad que presenta el sistema médico antes y después de la implementación de cada uno de los componentes que integran dicho módulo de seguridad, para realizar este proceso se asignaron valores entre 1 y 5 según el nivel de seguridad que se asigna al sistema médico de la ESPOCH.

Considerando que uno es el valor más bajo y cinco es el valor más alto de esta escala, en la tabla 2-4 se puede observar el criterio de evaluación asignado.

Tabla 2-4: Criterio de evaluación general.

CRITERIO	ESCALA DE EVALUACIÓN				
Valor	1	2	3	4	5
Índice	0.1	0.2	0.6	0.8	1.0
Cualitativa	Escasa seguridad	Poco nivel de seguridad	Medio nivel de seguridad	Alto nivel de seguridad	Excelente nivel de seguridad
Porcentaje	1%	25%	50%	75%	100%

Realizado por: Centeno, Hernán, 2024

4.3.2. Valoración de: Análisis de los datos que ingresa el usuario antes del módulo de seguridad

Antes de la integración del módulo de seguridad el sistema permitía el ingreso de cualquier tipo de datos para realizar consultas dentro del mismo, por esta razón se le asigna los siguientes valores para este indicador.

Tabla 3-4: Valoración de: Análisis de los datos que ingresa el usuario antes del módulo de seguridad

NÚMERO	VALOR	ÍNDICE	PORCENTAJE
Prueba 1	1	0.1	1%
Prueba 2	1	0.1	1%
Prueba 3	1	0.1	1%
Prueba 4	1	0.1	1%
Total	4	0.4	4%
Promedio	1	0.1	1%

Realizado por: Centeno, Hernán, 2024

4.3.3. Valoración de: Análisis de los datos que ingresa el usuario después del módulo de seguridad

Con la implementación del módulo de seguridad todos los datos que son ingresados al sistema por el usuario deben pasar por un filtro el mismo que determina si son datos confiables o se trata de código malicioso, para este indicador se asigna los siguientes valores.

Tabla 4-4: Valoración de: Análisis de los datos que ingresa el usuario después del módulo de seguridad

NÚMERO	VALOR	ÍNDICE	PORCENTAJE
Prueba 1	4	0.8	75%
Prueba 2	5	1.0	100%
Prueba 3	5	1.0	100%
Prueba 4	5	1.0	100%
Total	19	3.8	375%
Promedio	4.75	0.95	93.75%

Realizado por: Centeno, Hernán, 2024

Tabla resumen del primer criterio de evaluación.

Tabla 5-4: Tabla resumen del primer criterio de evaluación.

CRITERIO	VALOR	ÍNDICE
Con el módulo de seguridad	4.75	0.95
Sin el módulo de seguridad	1	0.1

Realizado por: Centeno, Hernán, 2024

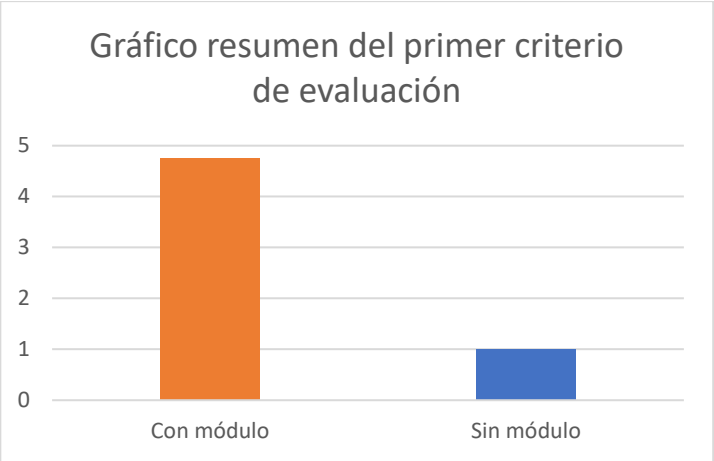


Gráfico 1-4: Resumen del primer criterio de evaluación

Realizado por: Centeno, Hernán, 2024

4.3.4. Valoración de: Registro de la actividad que realiza el usuario antes del módulo de seguridad.

Antes de la integración del módulo de seguridad el sistema no tenía un registro de las actividades que realiza cada usuario dentro del sistema, por esta razón se le asigna los siguientes valores para este indicador.

Tabla 6-4: Valoración de: Registro de la actividad que realiza el usuario antes del módulo de seguridad

NÚMERO	VALOR	ÍNDICE	PORCENTAJE
Prueba 1	2	0.2	25%
Prueba 2	1	0.1	1%
Prueba 3	1	0.1	1%
Prueba 4	1	0.1	1%
Total	5	0.5	28%
Promedio	1.25	0.125	7%

Realizado por: Centeno, Hernán, 2024

4.3.5. Valoración de: Registro de la actividad que realiza el usuario después del módulo de seguridad.

Con la integración del módulo de seguridad en la base de datos se creó una tabla de auditoría que permite el registro de las actividades que realizan los usuarios mientras está haciendo uso del sistema médico, para este indicador se asigna los siguientes valores.

Tabla 7-4: Valoración de: Registro de la actividad que realiza el usuario después del módulo de seguridad

NÚMERO	VALOR	ÍNDICE	PORCENTAJE
Prueba 1	4	0.8	75%
Prueba 2	5	1.0	100%
Prueba 3	5	1.0	100%
Prueba 4	5	1.0	100%
Total	19	3.8	375%
Promedio	4.75	0.95	93.75%

Realizado por: Centeno, Hernán, 2024

Tabla resumen del segundo criterio de evaluación.

Tabla 8-4: Tabla resumen del segundo criterio de evaluación

CRITERIO	VALOR	ÍNDICE
Con el módulo de seguridad	4.75	0.95
Sin el módulo de seguridad	1.25	0.125

Realizado por: Centeno, Hernán, 2024

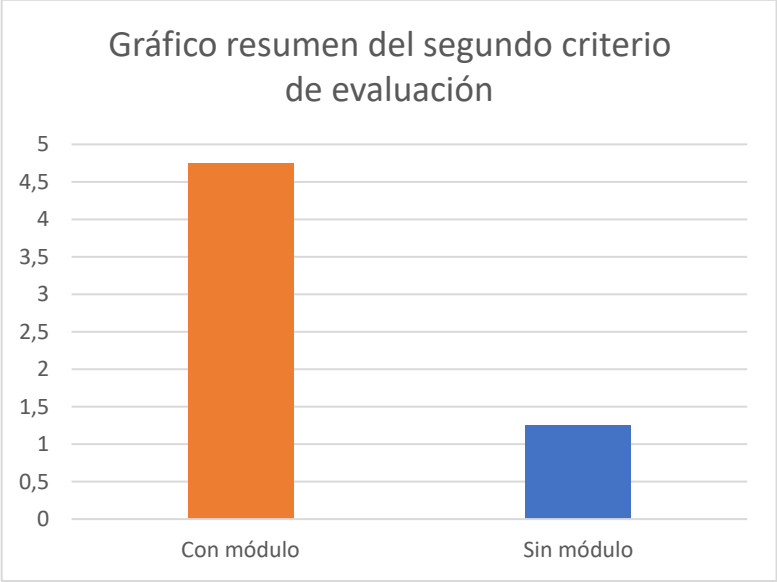


Gráfico 2-4: Resumen del segundo criterio de evaluación

Realizado por: Centeno, Hernán, 2024

4.3.6. Valoración de: Certificados SSL antes del módulo de seguridad.

Antes de la integración del módulo de seguridad el sistema médico fue puesto en producción haciendo uso del protocolo HTTP es decir conexión no segura, por esta razón se le asigna los siguientes valores para este indicador.

Tabla 9-4: Valoración de: Certificados SSL antes del módulo de seguridad

NÚMERO	VALOR	ÍNDICE	PORCENTAJE
Prueba 1	1	0.1	1%
Prueba 2	1	0.1	1%
Prueba 3	1	0.1	1%
Prueba 4	1	0.1	1%
Total	4	0.4	4%
Promedio	1	0.1	1%

Realizado por: Centeno, Hernán, 2024

4.3.7. Valoración de: Certificados SSL después del módulo de seguridad.

Con el objetivo de tener una transferencia de datos segura se instaló certificados SSL en los servidores de la vista y servicios del sistema médico, de esta manera el sistema utiliza el protocolo HTTPS, para este indicador se asigna los siguientes valores.

Tabla 10-4: Valoración de: Certificados SSL después del módulo de seguridad

NÚMERO	VALOR	ÍNDICE	PORCENTAJE
Prueba 1	4	0.8	75%
Prueba 2	4	0.8	75%
Prueba 3	5	1.0	100%
Prueba 4	5	1.0	100%
Total	18	3.6	350%
Promedio	4.5	0.9	87.5%

Realizado por: Centeno, Hernán, 2024

Tabla resumen del tercer criterio de evaluación.

Tabla 11-4: Resumen del tercer criterio de evaluación

CRITERIO	VALOR	ÍNDICE
Con el módulo de seguridad	4.5	0.9
Sin el módulo de seguridad	1	0.1

Realizado por: Centeno, Hernán, 2024

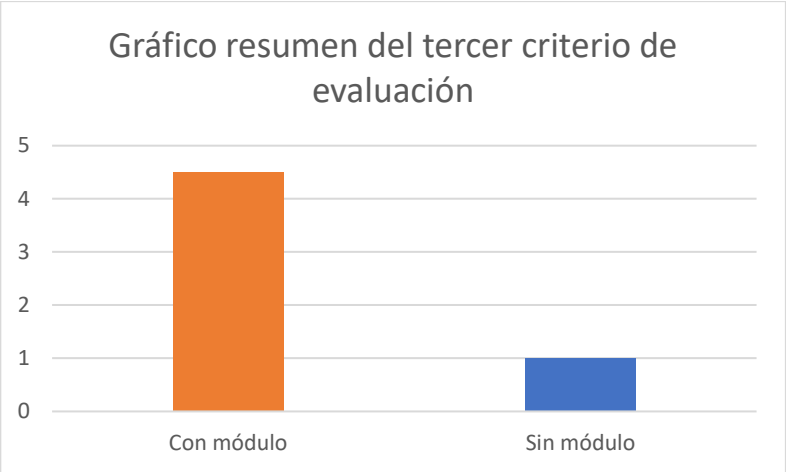


Gráfico 3-4: Resumen del tercer criterio de evaluación.

Realizado por: Centeno, Hernán, 2024

4.3.8. Valoración de: Copias de seguridad de la base de datos antes del módulo de seguridad.

Antes de la integración del módulo de seguridad no se tenía un criterio o política para realizar respaldos de la base de datos del sistema médico por lo que no se podía tener una copia de la información en caso de pérdida de datos, por esta razón se le asigna los siguientes valores para este indicador.

Tabla 12-4: Valoración de: Copias de seguridad de la base de datos antes del módulo de seguridad.

NÚMERO	VALOR	ÍNDICE	PORCENTAJE
Prueba 1	1	0.1	1%
Prueba 2	1	0.1	1%
Prueba 3	1	0.1	1%
Prueba 4	1	0.1	1%
Total	4	0.4	4%
Promedio	1	0.1	1%

Realizado por: Centeno, Hernán, 2024

4.3.9. Valoración de: Copias de seguridad de la base de datos después del módulo de seguridad.

Con la integración del módulo de seguridad se establecen políticas para la creación de copias de seguridad de la base de datos del sistema médico, para lo cual en el servidor de base de datos se ejecuta un script que automatice este proceso, para este indicador se asigna los siguientes valores.

Tabla 13-4: Valoración de: Copias de seguridad de la base de datos después del módulo de seguridad

NÚMERO	VALOR	ÍNDICE	PORCENTAJE
Prueba 1	4	0.8	75%
Prueba 2	4	0.8	75%
Prueba 3	5	1.0	100%
Prueba 4	5	1.0	100%
Total	18	3.6	350%
Promedio	4.5	0.9	87.5%

Elaborado por: **Centeno Hernán, 2024**

Tabla 14-4: Tabla resumen del primer criterio de evaluación

CRITERIO	VALOR	ÍNDICE
Con el módulo de seguridad	4.5	0.9
Sin el módulo de seguridad	1	0.1

Realizado por: Centeno, Hernán, 2024

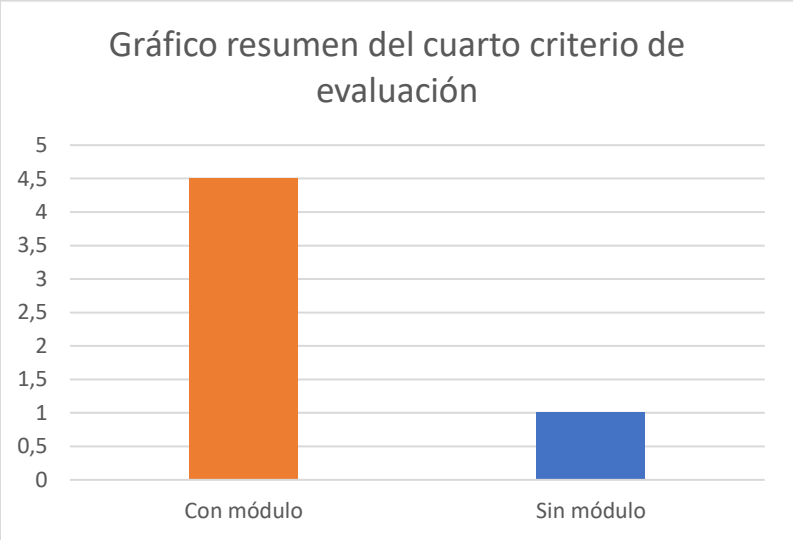


Gráfico 4-4: Resumen del cuarto criterio de evaluación.

Realizado por: Centeno, Hernán, 2024

4.3.9.1. Tabla de resultados globales de los datos obtenidos

Luego de realizar una evaluación de manera individual cada uno de los componentes que integran el módulo de seguridad para sistemas web se realiza una tabla de resultados totales globales en la que se refleja los valores obtenidos de la evaluación de cada componente, en la tabla 15-4 y en la tabla 16-4 se puede observar los datos obtenidos antes y después de la integración de dicho módulo de seguridad respectivamente.

Tabla 15-4: Resultado global de los criterios de evaluación previo a la integración del módulo de seguridad

COMPONENTE	VALOR	ÍNDICE
Análisis de los datos que ingresa el usuario.	1.0	0.1
Registro de la actividad que realiza el usuario.	1.25	0.125
Certificados SSL.	1.0	0.1
Copias de seguridad de la base de datos.	1.0	0.1
TOTAL	4.25	0.425
PROMEDIO	1.06	0.10

Realizado por: Centeno, Hernán, 2024

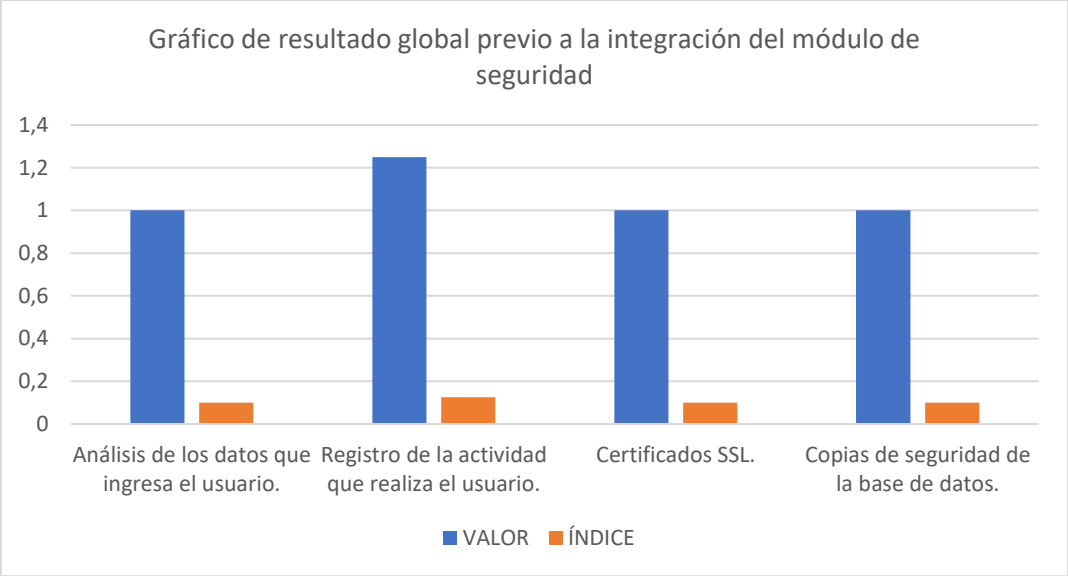


Gráfico 5-4: Resultado global previo la integración del módulo de seguridad

Realizado por: Centeno, Hernán, 2024

Tabla 16-4: Resultado globa de los criterios de evaluación con la integración del módulo de seguridad

COMPONENTE	VALOR	ÍNDICE
Análisis de los datos que ingresa el usuario.	4.75	0.95
Registro de la actividad que realiza el usuario.	4.75	0.95
Certificados SSL.	4.5	0.9
Copias de seguridad de la base de datos.	4.5	0.9
TOTAL	18.5	3.7
PROMEDIO	4.62	0.92

Realizado por: Centeno, Hernán, 2024

Gráfico del resultado global de los criterios de evaluación previo a la integración del módulo de seguridad.

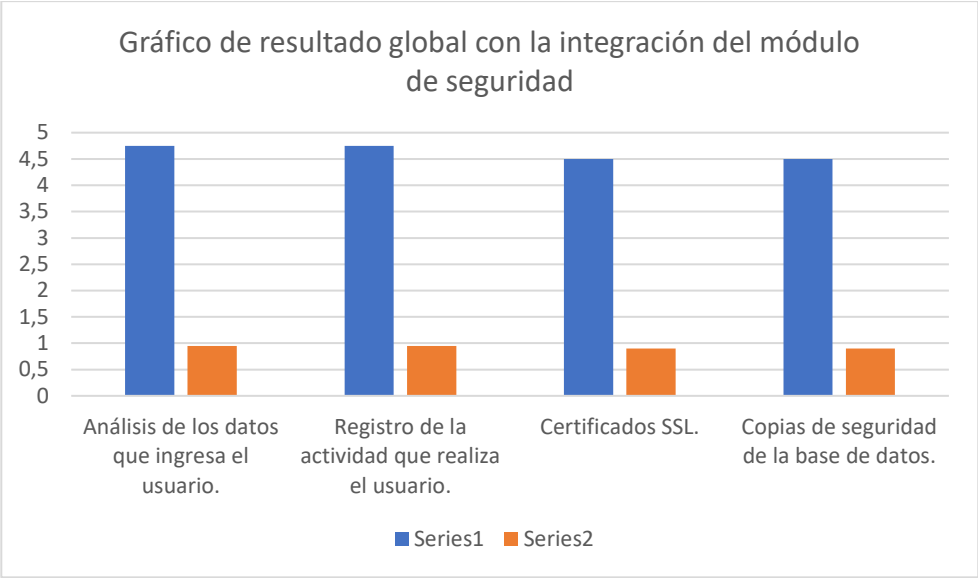


Gráfico 6-4: Resumen global con la integración del módulo de seguridad

Realizado por: Centeno, Hernán, 2024

En la tabla 17-4 se muestra de manera consolidada de la tabla 15-4 y la tabla 16-4, es decir hace referencia a los resultados obtenidos en los ambientes de evaluación realizados que son con la integración el módulo de seguridad y previo la integración de dicho módulo.

Tabla 17-4: Resultado globa de los criterios de evaluación con la integración del módulo de seguridad

COMPONENTE	CRITERIO	VALOR	ÍNDICE
Análisis de los datos que ingresa el usuario.	Con Mod.	4.75	0.95
	Sin Mod.	1.00	0.10
Registro de la actividad que realiza el usuario.	Con Mod.	4.75	0.95
	Sin Mod.	1.25	0.125
Certificados SSL.	Con Mod.	4.50	0.90
	Sin Mod.	1.00	0.10
Copias de seguridad de la base de datos.	Con Mod.	4.50	0.90
	Sin Mod.	1.00	0.10

Realizado por: Centeno, Hernán, 2024

4.4. T-Student

Según (Sánchez Turcios, 2015) la prueba t-Student se fundamenta en dos premisas; la primera: en la distribución de normalidad, y la segunda: en que las muestras sean independientes. Permite comparar muestras, $N \leq 30$ y/o establece la diferencia entre las medias de las muestras. El análisis matemático y estadístico de la prueba con frecuencia se minimiza para $N > 30$, utilizando pruebas no paramétricas, cuando la prueba tiene suficiente poder estadístico.

4.4.1. Total, de índices obtenidos de las pruebas realizadas.

En esta sección se presenta una tabla general con los valores totales de los índices obtenidos en cada uno de los escenarios de pruebas realizados en el sistema e-Salud de la ESPOCH con y sin el módulo de seguridad desarrollado.

Los escenarios de prueba planteados son los siguientes:

- **Escenario 1:** Análisis de los datos que ingresa el usuario.
- **Escenario 2:** Registro de la actividad que realiza el usuario.
- **Escenario 3:** Certificados SSL.
- **Escenario 4:** Copias de seguridad de la base de datos.

Tabla 18-4: Total, índices por prueba sin el módulo de seguridad

Prueba	Escenario de pruebas			
	Escenario 1	**Escenario 2**	**Escenario 3**	**Escenario 4**
Prueba 1	0.1	0.2	0.1	0.1
Prueba 2	0.1	0.1	0.1	0.1
Prueba 3	0.1	0.1	0.1	0.1
Prueba 4	0.1	0.1	0.1	0.1
X con módulo	**0.10**	**0.13**	**0.10**	**0.10**

Realizado por: Centeno, Hernán, 2024

Tabla 19-4: Total, índices por prueba con el módulo de seguridad

Prueba	Escenario de pruebas			
	Escenario 1	**Escenario 2**	**Escenario 3**	**Escenario 4**
Prueba 1	0.8	0.8	0.8	0.8
Prueba 2	1.0	1.0	0.8	0.8
Prueba 3	1.0	1.0	1.0	1.0
Prueba 4	1.0	1.0	1.0	1.0
X sin módulo	**0.95**	**0.95**	**0.90**	**0.90**

Realizado por: Centeno, Hernán, 2024

4.4.2. Comparación de los índices con el módulo de seguridad y sin el módulo de seguridad

Tabla 20-4: Índice de eficiencia del módulo de seguridad web

Comparación	Escenario de pruebas			
	Escenario 1	**Escenario 2**	**Escenario 3**	**Escenario 4**
Sin el módulo de seguridad	**0.10**	**0.13**	**0.10**	**0.10**
Con el módulo de seguridad	**0.95**	**0.95**	**0.90**	**0.90**
X sin módulo	0.11			
X con módulo	0.93			
SD sin módulo	0.0125			
SD con módulo	0.0289			

Realizado por: Centeno, Hernán, 2024

Después de haber obtenido los valores correspondientes al promedio (X) y desviación estándar (SD), se procede hacer una primera aproximación para los promedios muestrales, es decir se construye los intervalos de confianza alrededor de ambos promedios como se muestra en la tabla 21-4

Tabla 21-4: Intervalos de confianza

Sin el módulo de seguridad		**Con el módulo de seguridad**	
0.11 + 0.0125	0.11 - 0.0125	0.93 + 0.0289	0.93 - 0.0289
0.119	0.094	0.954	0.896

Realizado por: Centeno, Hernán, 2024

Sin la integración del módulo de seguridad web la media (X) obtenida de todos los índices como resultado de todos los escenarios de pruebas realizados es de 0,11 con una desviación estándar (SD) igual a: 0,0125 con un límite inferior de 0,094 y un límite superior igual a: 0,119. Con la integración del módulo de seguridad web la media (X) obtenida es de 0,93 con una desviación estándar (SD) de 0.0289 con un límite inferior de 0,896 y un límite superior de 0,954.

4.5. Planteamiento de la hipótesis

Para evaluar el funcionamiento del módulo de seguridad desarrollado se establecieron dos hipótesis que son la hipótesis nula (H0) y la hipótesis alterna (H1), en ese escenario la hipótesis alterna es la negación de la hipótesis nula.

4.6. Análisis de la hipótesis

Hipótesis nula:
En este caso de estudio como hipótesis nula se plantea que: las medias obtenidas con y sin la implementación del módulo de seguridad desarrollado son diferentes, el intervalo de confianza aplicado es del 95%, es decir se tiene la siguiente formula:

$$\mathbf{H0:}\ \mu ics = \mu icp$$

Hipótesis alterna:
En este caso de estudio como hipótesis nula se plantea que: las medias obtenidas con y sin la implementación del módulo de seguridad desarrollado son diferentes, el intervalo de confianza aplicado es del 95%, es decir se tiene la siguiente formula:

$$\mathbf{H1:}\ \mu ics \neq \mu icp$$

4.7. Prueba de T-Student

Tabla 22-4: Prueba de T-Student

Prueba t para medias de dos muestras emparejadas		
Resultados	**Sin el módulo de seguridad**	**Con el módulo de seguridad**
Media	0,10625	0,925
Varianza	0,00015625	0,00083333
Observaciones	4	4
Coeficiente de correlación de Pearson	0,577350269	
Diferencia hipotética de las medias	0	
Grados de libertad	3	
Estadístico t	-68,41251879	
P(T<=t) una cola	3,44112E-06	
Valor crítico de t (una cola)	2,353363435	
P(T<=t) dos colas	6,88224E-06	
Valor crítico de t (dos colas)	3,182446305	

Realizado por: Centeno, Hernán, 2024

4.7.1. Interpretación

El resultado del promedio de los índices de eficiencias con la integración del módulo de seguridad es significativamente mayor a comparación sin la integración del módulo de seguridad, de donde se obtiene lo siguiente:

Sin la integración del módulo de seguridad:	X = 0.11	SD = 0.0125
Con la integración del módulo de seguridad:	X = 0.925	SD = 0.0289

La sumatoria de el valor critico de T (una cola) que es igual a 2.3533 más el valor critico T (dos colas) que es igual a 3.1824 da un total de 5.5357, dicho resultado en comparación con valor del Estadístico T que es igual a -68.4125 es altamente significativa.

De igual manera al realizar la sumatoria del valor de P(T<=t) una cola que es igual a 3.44E-06 más el valor de P(T<=t) dos colas que es igual a 6.88E-06 se obtiene como resultado 1.03E-05, este valor obtenido se sumamente menor a 0.05, cabe recalcar que para este caso de estudio se trabajó con un nivel de confianza del 95%.

Por los datos obtenidos y expuestos anteriormente se evidencia que los promedios analizados son significativamente diferentes, razón por la cual se rechaza la hipótesis nula y acepta la hipótesis

alterna lo que significa que: “Con la implementación de un módulo de seguridad se mejora la integridad de la información del sistema de E-salud de la ESPOCH”.

CAPÍTULO V

5. PROPUESTA

En el este capítulo se presenta la propuesta realizada de un módulo de seguridad aplicado al sistema de Salud de la ESPOCH.

5.1. IMPLEMENTACIÓN DEL MÓDULO DE SEGURIDAD

En esta sección se describe el despliegue y funcionamiento del módulo de seguridad implementado en el sistema médico de la ESPPCH.

5.2. Registro de la actividad que realiza el usuario.

En esta opción se lleva un registro de la actividad que realiza cada usuario en el sistema e-Salud de la ESPOCH.

Para llevar a cabo esta opción se procedió a realizar el script de la tabla a ser implementada con las columnas que albergaría la información que permita llevar un control de las actividades de los usuarios, en la figura 1-5 se muestra la estructura de la tabla de auditoria.

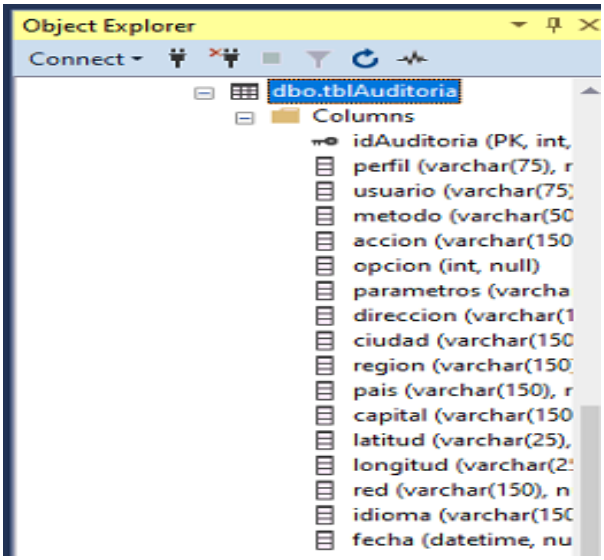
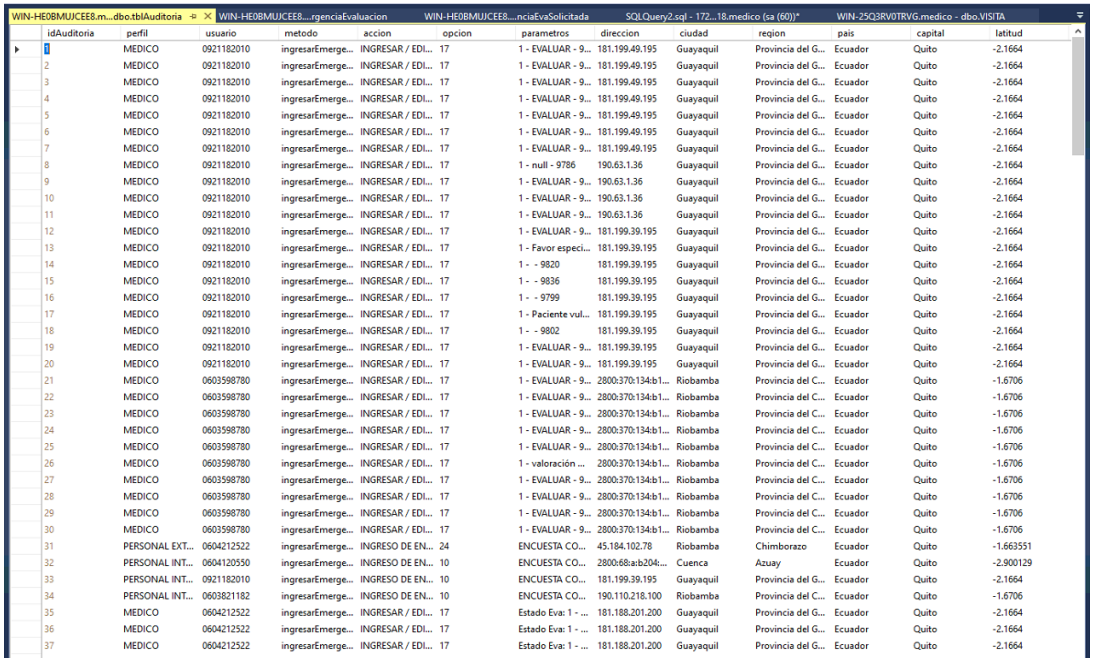


Figura 1-5: Estructura de la tabla de auditoria

Realizado por: Centeno, Hernán, 2024

En la figura 2-5 se muestra la información que aloja la tabla de auditoría, se almacena información como: el Id del usuario y el perfil, la acción que realiza, la fecha y hora de acceso, entre otra.



idAuditoria	perfil	usuario	metodo	accion	opcion	parametros	direccion	ciudad	region	pais	capital	latitud
1	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	181.199.49.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
2	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	181.199.49.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
3	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	181.199.49.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
4	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	181.199.49.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
5	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	181.199.49.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
6	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	181.199.49.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
7	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	181.199.49.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
8	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - null - 9786	190.63.1.36	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
9	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	190.63.1.36	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
10	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	190.63.1.36	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
11	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	190.63.1.36	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
12	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	181.199.39.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
13	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - Favor especi...	181.199.39.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
14	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - - 9820	181.199.39.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
15	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - - 9836	181.199.39.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
16	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - - 9799	181.199.39.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
17	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - Paciente vul...	181.199.39.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
18	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - - 9802	181.199.39.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
19	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	181.199.39.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
20	MEDICO	0921182010	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	181.199.39.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
21	MEDICO	0603598780	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	2800:370:134:b1...	Riobamba	Provincia del C...	Ecuador	Quito	-1.6706
22	MEDICO	0603598780	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	2800:370:134:b1...	Riobamba	Provincia del C...	Ecuador	Quito	-1.6706
23	MEDICO	0603598780	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	2800:370:134:b1...	Riobamba	Provincia del C...	Ecuador	Quito	-1.6706
24	MEDICO	0603598780	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	2800:370:134:b1...	Riobamba	Provincia del C...	Ecuador	Quito	-1.6706
25	MEDICO	0603598780	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	2800:370:134:b1...	Riobamba	Provincia del C...	Ecuador	Quito	-1.6706
26	MEDICO	0603598780	ingresarEmerge...	INGRESAR / EDI...	17	1 - valoración ...	2800:370:134:b1...	Riobamba	Provincia del C...	Ecuador	Quito	-1.6706
27	MEDICO	0603598780	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	2800:370:134:b1...	Riobamba	Provincia del C...	Ecuador	Quito	-1.6706
28	MEDICO	0603598780	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	2800:370:134:b1...	Riobamba	Provincia del C...	Ecuador	Quito	-1.6706
29	MEDICO	0603598780	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	2800:370:134:b1...	Riobamba	Provincia del C...	Ecuador	Quito	-1.6706
30	MEDICO	0603598780	ingresarEmerge...	INGRESAR / EDI...	17	1 - EVALUAR - 9...	2800:370:134:b1...	Riobamba	Provincia del C...	Ecuador	Quito	-1.6706
31	PERSONAL EXT...	0604212522	ingresarEmerge...	INGRESO DE EN...	24	ENCUESTA CO...	45.184.102.78	Riobamba	Chimborazo	Ecuador	Quito	-1.663551
32	PERSONAL INT...	0604120550	ingresarEmerge...	INGRESO DE EN...	10	ENCUESTA CO...	2800:68:a:b204:...	Cuenca	Azuay	Ecuador	Quito	-2.900129
33	PERSONAL INT...	0921182010	ingresarEmerge...	INGRESO DE EN...	10	ENCUESTA CO...	181.199.39.195	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
34	PERSONAL INT...	0603821182	ingresarEmerge...	INGRESO DE EN...	10	ENCUESTA CO...	190.110.218.100	Riobamba	Provincia del C...	Ecuador	Quito	-1.6706
35	MEDICO	0604212522	ingresarEmerge...	INGRESAR / EDI...	17	Estado Eva: 1 - ...	181.188.201.200	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
36	MEDICO	0604212522	ingresarEmerge...	INGRESAR / EDI...	17	Estado Eva: 1 - ...	181.188.201.200	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664
37	MEDICO	0604212522	ingresarEmerge...	INGRESAR / EDI...	17	Estado Eva: 1 - ...	181.188.201.200	Guayaquil	Provincia del G...	Ecuador	Quito	-2.1664

Figura 2-5: Información de la tabla de auditoria

Realizado por: Centeno, Hernán, 2024

5.3. Certificados SSL.

Los certificados SSL se implementaron en los servidores de producción tanto de la vista como de los servicios utilizados en el sistema e-Salud de la ESPOCH.

Para este proceso los certificados SSL fueron copiados a los servidores de producción tanto de la vista como de servicios y posteriormente será vinculado el certificado con el sitio web, para lo cual se debe seleccionar el certificado al crear el sitio web (figura 3-5) y establecer las credenciales de autenticación (figura 4-5).

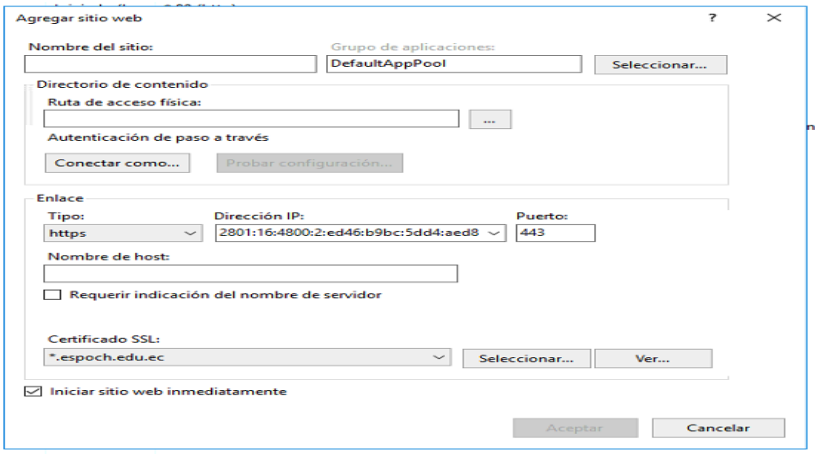


Figura 3-5: Creación del sitio y agregar el certificado SSL.

Realizado por: Centeno, Hernán, 2024

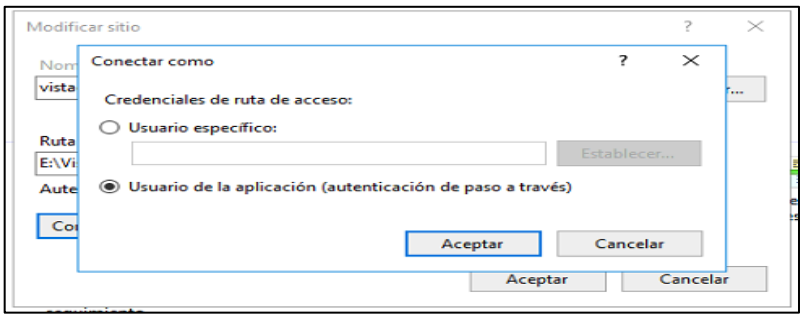


Figura 4-5: Establecer certificados SSL

Realizado por: Centeno, Hernán, 2024

En la figura 5-5 muestra los puertos por el cual se podrá acceder al sistema web ya sea por HTTP o HTTP.

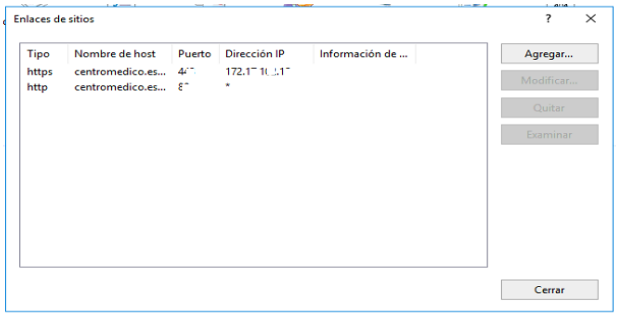


Figura 5-5: Puertos del sistema

Realizado por: Centeno, Hernán, 2024

5.4. Copias de seguridad de la base de datos

Este proceso permite realizar copias de seguridad o backup de la base de datos del sistema e-Salud de la ESPOCH.

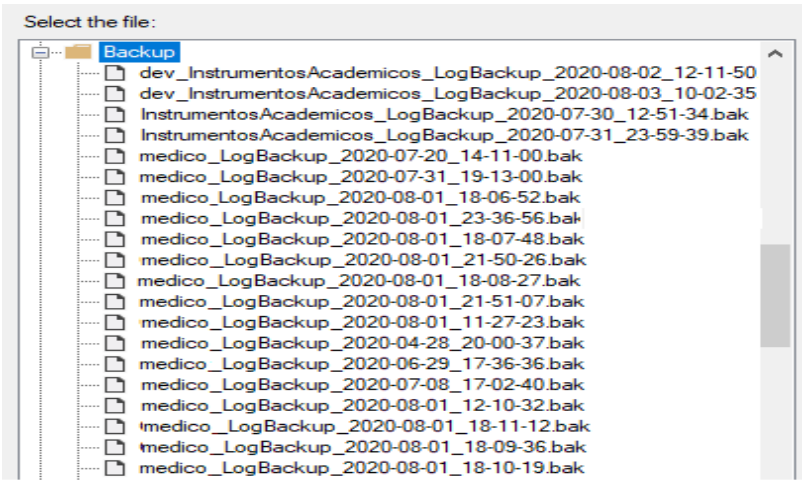


Figura 6-5: Copias de seguridad de la base de datos

Realizado por: Centeno, Hernán, 2024

CONCLUSIONES

Del presente trabajo realizado se concluye lo siguiente:

- Se logró desarrollar un módulo de seguridad el mismo que está constituido por cuatro componentes y a la vez se clasifican en dos grupos de acuerdo con la funcionalidad que cumple dentro del sistema médico, de esta manera los componentes del primer grupo hacen referencia a la interacción directa dentro del sistema médico y los componentes que corresponden al segundo grupo tienen interacción con los servidores de producción en donde se encuentra alojado el sistema.

- Para el desarrollo de los componentes que tienen interacción directa con el sistema médico se hizo uso del Framework Angular para la codificación, de esta manera se desarrolló una capa la misma que se encuentra ubicada entre los servicios que consume dentro del sistema y la codificación, de esta manera se tiene una capa intermedia la que se encarga de controlar los datos que ingresa el usuario en el sistema.

- Para la implementación de los componentes que interactúan directo con los servidores de producción se debe considerar que para el despliegue del sistema se utilizó tres servidores que son: de bases de datos, servicios web y de interfaz, en los tres servidores se hizo la implementación de certificados SSL de esta manera las peticiones se hacen por HTTPS.

- Para la evaluación del módulo de seguridad implementado se levantaron cuatro ambientes de pruebas es decir uno por cada componente del módulo desarrollado, y por medio del método estadístico de T-Student se hizo el análisis de los datos obtenidos, para este caso de estudio se trabajó con un nivel de confianza del 95%, con un nivel de significancia igual a 0.05, como resultado del estadístico T se obtiene un valor de P(T<=t) dos colas igual a 6.88224E-06 que en comparación al nivel de significancia se aprecia una diferencia significativa con los que se comprueba que la implementación del módulo desarrollado permite mejorar la seguridad en el sistema de salud de la ESPOCH.

RECOMENDACIONES

Luego de haber realizado el presente trabajo se recomienda lo siguiente:

- Se debe establecer normas y políticas de seguridad durante el desarrollo de sistemas informáticos ya sea que estén orientados para la web, sistemas de escritorio o sistemas para dispositivos móviles, que hagan consumo de servicios en la nube.

- Se debe tener un registro de las actividades que realiza cada usuario dentro de los sistemas, en el cual se registre la fecha hora y acción realizan los usuarios, de esta manera se podrá identificar un control de acceso a la información que alberga cada sistema.

- Se debería implementar políticas de seguridad a nivel de servidores y realizar un monitoreo contante para identificar posibles ataques y peticiones que se realizan, logrando obtener un informe o bitácora de ataques que reciben los servidor.

GLOSARIO

Visual Code: Editor de texto utilizado para escribir el código para realizar el módulo de seguridad web**.**

SQL server: Motor de base de datos usado para el almacenamiento de la información que registra el módulo de seguridad web**.**

Angular: Framework en el cual se desarrolla el módulo de seguridad web**.**

Windows Server: Servidor en donde se va alojar el módulo de seguridad web, el mismo que debe ser configurado para que se ejecute la aplicación**.**

BIBLIOGRAFÍA

Alva Obeso, M. E. (2005). *Metodología de Medición y Evaluación de la Usabilidad en Sitios Web Educativos*. Retrieved from http://di002.edv.uniovi.es/~cueva/investigacion/tesis/Elena.pdf

ALVEAR REINOSO, F. X. (2019). ANÁLISIS Y DISEÑO DE UNA PROPUESTA PARA MITIGAR ATAQUES CIBERNÉTICOS A CORREOS ELECTRÓNICOS UTILIZANDO TÉCNICAS DE HACKING ÉTICO. *Tesis*, *04*, 1–100. Retrieved from http://dspace.ups.edu.ec/bitstream/123456789/5081/1/UPS-CYT00109.pdf

Arteaga, F. (2019). *Ciberseguridad y seguridad integral en el sector energético*. 1–9.

Sosa Sosa, V. (2012). *Servicios Web con Software Libre*. 1-35

Cangás, F. X. C. (2015). *Sistema web transur con node.js para la gestión de transporte de la cooperativa de transporte de pasajeros inter cantonal urcuquí.* (Universidad Técnica del Norte). Retrieved from http://repositorio.utn.edu.ec/handle/123456789/4630

Cañón Parada, L. J. (2015). *Ataques Informáticos, Ethical Hacking*. Retrieved from http://polux.unipiloto.edu.co:8080/00002427.pdf

Cardenal, G. (2013). Ataques de inyección SQL. *HostaliaWhitepapers*, *1*, 48008–48902. Retrieved from http://pressroom.hostalia.com/wp-content/themes/hostalia_pressroom/images/inyeccion-sql-wp-hostalia.pdf

Carles, M. (2013). Desarrollo de aplicaciones web. In *Universitat Oberta de Catalunya Av.* (Vol. 112).

Castillo, C., López J., Arocena, F. (2011). MANUAL BÁSICO de creación de páginas web. *Area de La Tecnologia de La Informacion y Las Comunicaciones Aplicadas*, 57. Retrieved from https://www.um.es/atica/documentos/html.pdf

Centro Criptológico Nacional. (2019). Ciberamenazas y Tendencias. *Ciberamenazas y Tendencias.* Retrieved from https://www.ccn-cert.cni.es/informes/informes-ccn-cert-publicos/3776-ccn-cert-ia-13-19-ciberamenazas-y-tendencias-edicion-2019-1/file.html

Chalabe Jimenez, N. S. (2019). HACKING WEB (ANÁLISIS DE ATAQUES SQL Inyección, XSS). *Universidad Nacional Abierta y a Distancia - UNAD*, *23*(3), 2019.

Chavarria Neira, B., & Gudiño de la A, E. (2017). *IMPLEMENTACIÓN DE UN SERVIDOR WEB Y UN DISEÑO DE UNA PÁGINA UTILIZANDO HERRAMIENTAS DE SOFTWARE LIBRE PARA EL DISPENSARIO "SAGRADA FAMILIA" DE LA CIUDAD DE GUAYAQUIL.* Retrieved from https://dspace.ups.edu.ec/bitstream/123456789/14162/1/GT001840.pdf

Cienfuegos Velasco, M. de los A. (2019). Reflexiones en torno al método científico y sus etapas. *RICSH Revista Iberoamericana de Las Ciencias Sociales y Humanísticas*, *8*(15), 60–77. https://doi.org/10.23913/ricsh.v8i15.161

Cruz Gavilanes, Y., & Martínez Santander, C. (2017). Metodología OSSTMM para la detección de errores de seguridad y vulnerabilidad en sistemas operativos de 64 bits a nivel de usuario final. *Dominio de Las Ciencias*, *3*(3), 505–516. https://doi.org/10.23857/dom.cien.pocaip.2017.3.mono1.ago.505-516

Datacom. (2019). *LOS PRINCIPALES RETOS DE LA CIBERSEGURIDAD EN 2019*. 1–16.

Dussan Clavijo, C. A. (2006). POLÍTICAS DE SEGURIDAD INFORMÁTICA. In *Entramado* (Vol. 2). Retrieved from http://www.redalyc.org/html/2654/265420388008/

Fajardo, H. J. G., & Silva, E. A. L. (2014). Diseño de un modelo funcional de gestión soportado en servicios RESTful para gestión integrada de redes y servicios de T-learning. *Sistemas & Telemática*, Vol. 12, pp. 49–65. Retrieved from https://www.redalyc.org/pdf/4115/411533999003.pdf

Fernandes-Oruña, J. C., Sanchez Jimebes, F., Herrera Sanchez, D., Martínez Moreno, F., Rubio García, M., Gil Pérez, V., … Gómez Martín, M. A. (2019). *Estudio sobre la cibercriminalidad en España*. (1), 6–8. https://doi.org/10.16309/j.cnki.issn.1007-1776.2003.03.004

Fernández Rodríguez, J. C., Miralles Muñoz, F., & Millana Cuevas, L. (2019). Perfil psicosociológico en el ciberdelincuente. *RICSH Revista Iberoamericana de Las Ciencias Sociales y Humanísticas*, *8*(16), 156–177. https://doi.org/10.23913/ricsh.v8i16.179

Fernández Romero, Y., & Díaz González, Y. (2012). Patrón Modelo-Vista-Controlador. *Revista Telemática*, *11*(1), 47–57.

Foundation, O. (2017). OWASP Top 10 - 2017 Los diez riesgos más críticos en Aplicaciones Web. *Journal of Chemical Information and Modeling*.

Fuertes Maestro, A. (2014). *Elaboración de una metodología de test de intrusión dentro de la auditoria de seguridad*. Retrieved from https://reunir.unir.net/bitstream/handle/123456789/2331/AntonioFuertesMaestroTFM.pdf?sequence=3&isAllowed=y

Garrido, J. S. C. (2014). *Arquitectura y diseño de sistemas web modernos*. Retrieved from http://pegaso.ls.fi.upm.es/~sortega/html_css/files/Arquitectura_y_disenyo_de_sistemas_web_modernos.pdf

Gordon Revelo, D., & Pacheco Villamar, R. (2018). Análisis de Estrategias de Gestión de Seguridad Informática con Base en la Metodología Open Source Security Testing Methodology Manual (OSSTMM) para la Intranet de una Institución de Educación Superior. *Computación e Informática*, *7*(1), 1–21. Retrieved from http://repositorio.uees.edu.ec/handle/123456789/2410

GORDÓN REVELO, D. S. (2017). *Análisis de estrategias de gestión de seguridad informática con base en la metodología Open Source Security Testing Methodology Manual (OSSTMM) para la intranet de una Institución de Educación Superior*. 55.

Guerrero Ortega, J. C., Jiménez Toledo, R. A., Muñoz Guzmán, J. A., Zambrano Villota, A. M., & Ojeda Ortiz, G. A. (2013). *Aspectos fundamentales en la construcción de páginas web seguras basados en OWASP*. Retrieved from http://www.ojseditorialumariana.com/index.php/BoletinInformativoCEI/article/view/1416/1378

Infante Prieto, J. (2014). *Introducción a las Aplicaciones Web*. 2-9

Jalal Caál, J. C., Ramos Ramírez, M. R., Ajcuc Ortiz, A., Lorenty, C. R., & Diéguez Hernández, P. (2015). METODOS DE INVESTIGACION. *Universidad San Carlos De Guatemala, Facultad De Humanidades*, *1*(4), 53.

Abreu, J. (2012). Hipótesis, Método & Diseño de Investigación. In *Daena: International Journal of Good Conscience* (Vol. 7). Retrieved from http://www.spentamexico.org/v7-n2/7(2)187-197.pdf

Kiessling, M., & Junge, H. A. (2015). *El Libro para Principiantes en Node.js*. Retrieved from http://www.nodebeginner.org/index

Marini, E. (2014). *El Modelo Cliente/Servidor*. 1–11.

MARONE, L., & GALETTO, L. (2014). *El doble papel de las hipótesis en la investigación ecológica y su relación con el método hipotético-deductivo*. Retrieved from https://www.researchgate.net/publication/262596299

Marrero Travieso1, Y. (2003). *La Criptografía como elemento de la seguridad informática*. Retrieved from https://s3.amazonaws.com/academia.edu.documents/43689560/criptografia.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1538975893&Signature=oa%2Bfk3VSmDrrLgy1FNC1loIQPwA%3D&response-content-disposition=inline%3Bfilename%3DLa_Criptografia_como_elemento_de_

Mejías, Cintado, S. (2018). *CÓMO CONFIGURAR INTERNET INFORMATION SERVER*. Retrieved from http://s219540635.mialojamiento.es/feb2010/CONFIGURAR_INFORMATION_SERVER.pdf

Mestras, J. P. (2013). Servidores Web – Apache. *Creative Commons*, 1–36. Retrieved from https://www.fdi.ucm.es/profesor/jpavon/web/31-ServidoresWeb-Apache.pdf

Mieres, J. (2009). *Ataques informáticos Debilidades de seguridad comúnmente explotadas*. 17. Retrieved from https://www.evilfingers.net/publications/white_AR/01_Ataques_informaticos.pdf

Miranda, C. P. (2019). *LA METODOLOGÍA OSSTMM V3, PARA EL MINISTERIO DE INCLUSIÓN ECONÓMICA Y SOCIAL*.

Monago Ruiz, A. (2019). *Servicio Web API REST sobre el Framework Spring, Hibernate, JSON Web Token y BBDD Oracle*. University of Universidad de Sevilla; 2016-2021

Montenegro, M. (2016). *Introducción a las Tecnologías Web*. *467*, 86. Retrieved from http://dalila.sip.ucm.es/~manuel/JSW1/Slides/ServiciosWeb.pdf

Montesino Perurena, R., Baluja García, W., & Joelsy Porvén, R. (2013). Gestión automatizada e integrada de controles de seguridad informática. In *Ingeniería Electrónica, Automática y Comunicaciones* (Vol. 34). Retrieved from http://scielo.sld.cu/scielo.php?pid=S1815-59282013000100004&script=sci_arttext&tlng=pt

Mora Castillo, J. A. (2016). Serialización/deserialización de objetos y transmisión de datos con JSON: una revisión de la literatura. *Tecnología En Marcha, ISSN 0379-3962, ISSN-e 2215-3241, Vol. 29, Nº. 1, 2016, Págs. 118-125*, *29*(1), 118–125. Retrieved from https://dialnet.unirioja.es/servlet/articulo?codigo=5432058

Murillo, J., Alonso Serrano, A., García Sanz, L., León Rodrigo, I., García Gordo, E., Gil Álvaro, B., & Ríos Brea, L. (2012). Métodos de investigación de enfoque experimental. *Metodología de La Investigación Educativa*, 167–193. Retrieved from http://www.postgradoune.edu.pe/documentos/Experimental.pdf

Node.js. (2017). *Aprendizaje Node.js*. 30-35 Retrieved from https://riptutorial.com/es/home

Node, JS. (2017). JavaScript en el Servidor *Node . js*. 1-42

OWASP. (2018). Guía para evitar infecciones de Ransomware. *Recuperado El 25 de Abril de 2020 De :*, 1–53. Retrieved from https://owasp.org/www-pdf-archive/Owasp-guia-evitar-ransomware_es.pdf

Peña, J. M. (2016). *Protección de Datos y Backup Contenidos*. 1–29.

Pérez, V. (2015). Desarrollo De Un Sitio Web Para Un Colegio. *Universidad Politecnica de Valencia*.

Perurena Cancio, L., & Moráguez Bergues, M. (2013). Usabilidad de los sitios Web, los métodos y las técnicas para la evaluación. *Revista Cubana de Información En Ciencias de La Salud*, *2*, 19. Retrieved from https://www.medigraphic.com/pdfs/acimed/aci-2013/aci132g.pdf

Prieto Donate, F. (2014). Transmisión de imágenes de video mediante Servicios Web XML sobre J2ME. *Biblioteca de Ingeniería. Universidad de Sevilla*, 332. Retrieved from https://cutt.ly/fsueXUf

Quiroz Zambrano, S. M., & Macías Valencia, D. G. (2017). Seguridad en informática: consideraciones. In *Dominio de las Ciencias, ISSN-e 2477-8818, Vol. 3, Nº. Extra 3, 2017, págs. 676-688* (Vol. 3). Retrieved from https://dialnet.unirioja.es/servlet/articulo?codigo=6137824

Romero Castro, M. I., Figueroa Morán, G. L., Vera Navarrete, D. S., Álava Cruzatty, J. E., Parrales Anzúles, G. R., Álava Mero, C. J., … Castillo Merino, M. A. (2018). Introducción a la seguridad informática y el análisis de vulnerabilidades. In *Editorial Área de Innovación y Desarrollo,S.L.* https://doi.org/10.17993/ingytec.2018.46

Sahuquillo, C. (2011). *Test de Intrusión : Metodologías OSSTMM e ISSAF*.

Sánchez Turcios, R. A. (2015). Student's t. Uses and abuses. *Revista Mexicana de Cardiologia*, *26*(1), 59–61.

Echeverry Parada, J. (2009). *METODOLOGÍA PARA EL DIAGNÓSTICO CONTINUO DE LA SEGURIDAD INFORMÁTICA DE LA RED DE DATOS DE LA UNIVERSIDAD MILITAR NUEVA GRANADA*.

TAPUY CHONGO, S. S. (2019). *SISTEMA DE INFORMACIÓN CON ARQUITECTURA MVC PARA EL CONTROL DE INVENTARIO DE PRODUCTOS DE LA EMPRESA "DECOREY PUYO." 23*(3), 2019.

Toth, G. (2017). *Vulnerabilidades en aplicaciones web.* 25–34.

Alec, H., et al. (2017). *TypeScript*. *Ebook*, pp. 1–124, 2017-2021. Available from: https://riptutorial.com/es/home

Valbuena, Á. M. A. (2014). *Guía comparativa de frameworks para los lenguajes html 5, css y javascript para el desarrollo de aplicaciones web* (UNIVERSIDAD TECNOLÓGICA DE PEREIRA FACULTAD DE INGENIERÍAS). Retrieved from https://core.ac.uk/download/pdf/71397979.pdf

Valdez Alvarado, A. (2013a). Osstmm 3. *Revista de Información, Tecnología y Sociedad*, 29.

Valdez Alvarado, A. (2013b). *OSSTMM 3*. Retrieved from http://www.dragonjar.org/osstmm-manual-de-la-

Valverde Ramos, E., & Mora de Fuentes, P. H. (2017). *Introducción a TypeScript*. Retrieved from http://fanta.56k.es/libros/tecnicos/Manual-TypeScript.pdf

Vazques, J. (2017). *Protocolo Http*. *5*, 35–42. Retrieved from http://bibing.us.es/proyectos/abreproy/11214/fichero/TOMO+I%252F05+Capitulo+5+Protocolo+HTTP.pdf

Yeiniel, A., Briseida, B., & Liuba, L. M. (2012). Auditoría con Informática a Sistemas Contables. *Revista de Arquitectura e Ingeniería*, *6*(2), 1–14.

espoch | Dirección de Bibliotecas y Recursos del Aprendizaje

UNIDAD DE PROCESOS TÉCNICOS Y ANÁLISIS BIBLIOGRÁFICO Y DOCUMENTAL

REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: 08 / 11 / 2023

INFORMACIÓN DEL AUTOR/A (S)
Nombres – Apellidos: *Hernán Darío Centeno Aulla*
INFORMACIÓN INSTITUCIONAL
Instituto de Posgrado y Educación Continua
Título a optar: *Magíster en Seguridad Telemática*
f. Analista de Biblioteca responsable: Lic. Luis Caminos Vargas Mgs.

Firmado electrónicamente por:
LUIS ALBERTO
CAMINOS VARGAS

UNIDAD DE PROCESOS TÉCNICOS DBRA ESPOCH

00069-DBRA-UTP-IPEC-2023