**OFICINA DE POSGRADOS**

**Tema:**

# VULNERABILIDADES EN APLICACIONES WEB UTILIZANDO LA METODOLOGÍA DE "PROYECTO ABIERTO DE SEGURIDAD DE APLICACIONES WEB"

**Proyecto de investigación previo a la obtención del título de Magister en Ciberseguridad**

**Línea de Investigación:**

SEGURIDAD DE LA INFORMACIÓN

**Autor:**

DIEGO GAMBOA SAFLA

**Director:**

Ing. Mg. GALO LÓPEZ SEVILLA

**Ambato – Ecuador**

**Marzo 2021**

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR**

**SEDE AMBATO**

**HOJA DE APROBACIÓN**

Tema:

VULNERABILIDADES EN APLICACIONES WEB UTILIZANDO LA METODOLOGÍA DE "PROYECTO ABIERTO DE SEGURIDAD DE APLICACIONES WEB"

**Línea de Investigación:**

SEGURIDAD DE LA INFORMACIÓN

Autor:

DIEGO LEONARDO GAMBOA SAFLA

Paul Bernal Ing. MSc. f._____
CALIFICADOR

Marcelo Balseca Ing. Mg. f._____
CALIFICADOR

Galo López Sevilla Ing. Mg. f._____
CALIFICADOR

Lincoln Josue Tamayo del Rio, Ing. PhD. f._____
DIRECTOR UNIDAD ACADEMICA

Hugo Rogelio Altamirano Villaroel Dr. f._____
SECRETARIO GENERAL PUCESA

Ambato – Ecuador
Marzo 2021

## DECLARACIÓN Y AUTORIZACIÓN

Yo: DIEGO LEONARDO GAMBOA SAFLA, con CI. 1804688214, autora del trabajo de graduación intitulado: "VULNERABILIDADES EN APLICACIONES WEB UTILIZANDO LA METODOLOGÍA DE "PROYECTO ABIERTO DE SEGURIDAD DE APLICACIONES WEB"", previa a la obtención del título profesional de Magister en Ciberseguridad, en la Pontificia Universidad Católica del Ecuador Sede Ambato.

1.- Declaro tener pleno conocimiento de la obligación que tiene la Pontificia Universidad Católica del Ecuador, de conformidad con el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de graduación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la Pontificia Universidad Católica del Ecuador a difundir a través de sitio web de la Biblioteca de la PUCE Ambato, el referido trabajo de graduación, respetando las políticas de propiedad intelectual de Universidad

Ambato, marzo 2021

**DIEGO LEONARDO GAMBOA SAFLA**

**CI: 1804688214**

## AGRADECIMIENTO

En el presente Proyecto de Titulación, quiero hacer un extenso agradecimiento, a todos quienes conforman la Maestría en Ciberseguridad de la Pontificia Universidad Católica del Ecuador Sede Ambato – Primera Cohorte, que de una u otra manera hicieron posible que este trabajo previo a mi graduación se realice exitosamente, así también, quiero manifestar un agradecimiento especial a mi Tutor Ing. Mg. Galo López, que gracias a su guía y apoyo, se logró trazar este importante objetivo principal, que se encuentra plasmado en este documento.

## DEDICATORIA

Dedico este proyecto de titulación a mis padres y mi hermano, que siempre me apoyaron en todo momento, ya que, son un eje fundamental en mi vida, que, con esfuerzo, sacrificio y de una u otra manera fueron una guía importante en mi carrera profesional para poder cumplir este importante objetivo.

Dedico también este proyecto de titulación a mi amada novia, que siempre estuvo conmigo cuando más lo necesitaba, y demostrarme de manera incondicional su amor, cariño y comprensión, contar con su apoyo constantemente para poder alcanzar este importante éxito profesional en mi vida.

Dedico a toda mi familia y amigos que fueron participes para que mi proyecto de titulación pueda culminarse con éxito.

Y sobre todo a mi Diosito por guiarme, iluminarme, y protegerme, además, me brinda fuerzas para seguir adelante en todo momento y no permitirme que los problemas, que se me presentaron fueran obstáculos en cumplir mi meta, me ha enseñado a afrontar siempre las adversidades sin perder nunca la dignidad ni desfallecer en el intento.

## RESUMEN

En las aplicaciones web en la Universidad Técnica de Ambato se muestra información de vital importancia, en muchas ocasiones ésta información es estática y en otras dinámica; en todos los casos un portal web ofrece una ventana que ciberdelincuentes logran utilizar como un medio para un ataque; por eso resulta importante, realizar un análisis de vulnerabilidades de software que existen en las plataformas que brindan soporte en la Universidad Técnica de Ambato; para cumplir este objetivo, se aplica la metodología de PROYECTO ABIERTO DE SEGURIDAD DE APLICACIONES WEB (OWASP), que aporta diferentes enfoques para el análisis de vulnerabilidades que utiliza herramientas que ayudan a realizar pruebas de penetración en aplicaciones web, además, aplica métodos para resolución y mitigación de dichas vulnerabilidades. Este trabajo se realiza a una determinada aplicación web de la Institución, durante el segundo semestre del año académico 2020, para demostrar el cumplimiento de los objetivos, se utiliza diferentes guías de prueba de ataques a sitios web que utiliza herramientas de código abierto; con lo que se espera, que éste aporte de solución a la seguridad informática en la aplicación web de la Institución, obteniéndose así un conjunto de buenas prácticas en cuanto a la seguridad en aplicaciones web se refiere. La metodología OWASP aporta diferentes enfoques para el análisis de vulnerabilidades en aplicaciones web.

Palabras clave: Vulnerabilidades, Proyecto abierto de Seguridad de Aplicaciones Web (OWASP), Seguridad Informática.

## ABSTRACT

The web applications at the Technical University of Ambato depict significantly important information. This information is sometimes static, and at other times it is dynamic, but in all cases a web portal offers a window that cyber-delinquents can use as a means for attack. For this reason, it is important to analyze the existing software vulnerabilities within the platforms that offer support to the Technical University of Ambato. In order to meet this objective, the Open Web Application Security Project (OWASP) methodology was used due to the fact that it provides different approaches for the analysis of vulnerabilities using tools that test penetrability of web applications, as well as methods for resolving and mitigating such vulnerabilities. This project has been carried out on one of the web applications of the institution, during the second term of the 2020 academic year. In order to demonstrate the fulfilment of the objectives, several different guides for test attacks against web sites are used by means of open-source coding tools. In this way, it is hoped to provide solutions for IT security in the institution's web applications. OWASP methodology provides different approaches for the analysis of the vulnerabilities in web applications.

Keywords: vulnerabilities, Open Web Application Security (OWASP), IT security

# ÍNDICE

## Índice de Figuras

## Índice de Tablas

## INTRODUCCIÓN

En los últimos años, se ha visto reflejado intrusiones continuas a diferentes aplicaciones web ya sean estas, educativas, financieras o gubernamentales, mediante ataques a bases de datos y páginas web, son un blanco perfecto para los ciberdelincuentes, por lo que muchas aplicaciones presentan vulnerabilidades que aumenta continuamente, y su nivel de explotación es cada vez más impactante.

En el ámbito internacional, Willberg (2019) describe los riesgos de seguridad de aplicaciones web más comunes según la lista OWASP (Open Web Application Security Project) Top 10 - 2017. El objetivo era aumentar la seguridad de la aplicación web de destino que informa sobre los posibles riesgos de seguridad, que se descubrieron durante la prueba, de modo, que se puedan tomar acciones correctivas para mitigarlos. Por lo tanto, es imprescindible tomar acciones o medidas correctivas para tratar de solucionar o mitigar los problemas, que se pueden encontrar en una aplicación web.

Por otro lado Chavarria Gonzalez (2020) manifiesta que típicamente a la hora de desarrollar una aplicación web no se le da importancia a hacerla segura, normalmente, se ignoran las consecuencias de estas vulnerabilidades. Entre la documentación que genera OWASP, una asociación, se dedica en exclusiva a estudiar y generar herramientas para la securización de las aplicaciones web, existe un compendio de las diez vulnerabilidades más comunes en las aplicaciones web.

OWASP. (2020). OWASP Top Ten. https://owasp.org/www-project-top-ten/. Recuperado 30/11/2020, se indica los principales riesgos de seguridad a las aplicaciones web, las cuales indican las vulnerabilidades más comunes que se realizan a aplicaciones web, además en el documento presentado por OWASP, se analiza el impacto potencial de cada vulnerabilidad y cómo evitarlas, finalmente, se incluye una guía de mejores prácticas a tomar en cuenta para la seguridad de aplicaciones web.

En contraste con lo que se menciona en el ámbito internacional, en todo momento es necesario mantener una adecuada y correcta seguridad aplicada a sistemas web que dentro de una organización, se mantiene así un correcto funcionamiento de las mismas y asegurar que la información dentro de ellas sea de total confianza para el usuario final, que evita, que personas malintencionadas sustraigan información, que luego, se vea afectada la integridad y la disponibilidad por parte una determinada organización a sus clientes de la misma.

> Los piratas informáticos suelen poner sus miras en aquello que alberga más usuarios. Una manera de lograr tener más probabilidades en sus objetivos. Esto sin duda puede ocurrir en sitios y aplicaciones web. Pueden buscar posibles vulnerabilidades existentes para llevar a cabo sus ataques. Las aplicaciones web son servicios y funciones muy variados. Pueden ser, por ejemplo, las herramientas para iniciar sesión, proceso de compra en una página o funciones para administrar el contenido de una red social. Javier Jiménez. (20 de abril, 2020). Tipos de Ataques a aplicaciones web que debes conocer. RedesZone. https://www.redeszone.net/tutoriales/seguridad/tipos-ataques-aplicaciones-web.

En el contexto ecuatoriano, Intriago & Karina (2018) anuncia que al no existir una metodología de pruebas de penetración orientada al riesgo, el auditor puede estar limitado en su evaluación. De esta manera, con la nueva metodología implementada en esta investigación, se dotaría a los auditores de una herramienta más ágil para evaluar los mayores riesgos y priorizarlos, y además es desarrollar una propuesta de metodología para pruebas de penetración orientada a riesgos con base a las metodologías, Open Source Security Testing Methodology Manual [OSSTMM], Open Web Application Security Project [OWASP], Information Systems Security Assessment Framework [ISSAF], Penetration testing execution standard [PTES], Common Vulnerability Scoring System [CVSS] dentro del campo de la auditoría de tecnología de información. (p. 3)

Por lo expuesto, anteriormente, se podrá deducir que la Metodología OWASP, ayuda a resolver problemas de ataques en base a las vulnerabilidades que presentan las aplicaciones web, estas actualmente han sido un blanco perfecto para la toma y el control de un determinado sistema por parte de un atacante informático, y de esta manera verse afectadas a las organizaciones.

Según Serna & Andrés (2019) Cualquier organización que expone sus servicios informáticos a redes de acceso tendrán que realizar un esfuerzo significativo para asegurar que la información y recursos estén protegidos. Internet es un factor primordial en la comunicación, sin dejar a un lado, los riesgos potenciales que se tienen en los accesos o en el mal uso de los servicios e información disponibles. Obviamente, existen sistemas más críticos que otros donde su seguridad debe de ser más alta y muy significativa, pero en general todas las aplicaciones Web deben de estar protegidas y aseguradas ante los principales ataques.

Por lo expuesto anteriormente, se podrá recalcar que toda aplicación web, requiere una atención esencial en cuanto a seguridad, mediante alguna página web resultaría una entrada perfecta a la

infraestructura tecnológica de cualquier organización, de la cual, se obtendría información importante, que afectaría el funcionamiento de esta, que desencadenaría pérdidas económicas o tecnológicas.

> Las 10 principales vulnerabilidades según OWASP (Proyecto abierto de seguridad de aplicaciones web) por sus siglas en inglés, engloban los tipos de vulnerabilidades más frecuentes que se observan en las aplicaciones web. Para evitar la percepción errónea que suelen perpetuar los proveedores de seguridad, no constituyen una lista de comprobación de los vectores de ataque que pueden bloquearse simplemente a través de un firewall de aplicaciones web (WAF). En cambio, su objetivo es concienciar sobre las vulnerabilidades de seguridad más habituales que deben tener en cuenta los desarrolladores de aplicaciones, mejorar dicha concienciación en una serie de prácticas de desarrollo y ayudar a inculcar una cultura de desarrollo seguro. Akamai. (Recuperado 11/11/2020). Como mejorar con akamai las prácticas de seguridad para mitigar los 10 principales riesgos. https://www.akamai.com.

Por otro lado Zapata (2019) afirma que la seguridad en las aplicaciones debe controlarse desde la etapa de desarrollo, la falta de una metodología clara que facilite una guía para el desarrollo y una etapa concisa de pruebas sobre las aplicaciones antes de salir a producción permite que estas aplicaciones presenten fallos y vulnerabilidades que representan altos riesgos para la compañía (p. 3).

En correspondencia con lo anterior, el presente proyecto tiene como objetivo mostrar los resultados de una investigación que tiene como finalidad buscar las principales vulnerabilidades en una determinada aplicación web de la Universidad Técnica de Ambato.

En atención a la problemática expuesta para la investigación, se menciona que el determinar este aspecto es fundamental para establecer hasta qué punto es válido aplicar los diferentes procedimientos, herramientas y pruebas de seguridad que propone la metodología OWASP a ser aplicados a la aplicación web de la Universidad Técnica de Ambato, y toma en cuenta que los resultados obtenidos, contribuirán a diseñar estrategias más seguras para realizar pruebas periódicas de evaluaciones de seguridad a las aplicaciones web.

Debido al constante manejo de datos informativos a través de páginas web de la Universidad Técnica de Ambato , estos se ven inseguros, se conviertan en un blanco perfecto para los ciberdelincuentes; la falta de seguridad en las páginas web en cuanto a entrega de credenciales, o

datos que son estrictamente personales, hacen que las páginas requieran más seguridad en su acceso, y a su vez, que al momento de solicitar datos importantes, se tomen las seguridades necesarias para beneficio de los usuarios; además, se debería tomar medidas de protección a páginas web que sean de uso frecuente por usuarios que requieran utilizar plataformas con fines educativos y financieros.

En la Universidad Técnica de Ambato existe una diversidad de sitios web que brindan varios servicios, desde financieros, académicos y de comunicación. En todos ellos resulta un problema el manejo de la seguridad, pues, se manejan lenguajes de programación diversos, y en algunos casos, que son desarrollados por diferentes grupos de personas; esto genera un ecosistema de servicios web, en donde el manejo de la seguridad de los sitios se convierte en un problema. En este contexto el problema es: ¿Cómo implementar mejor los estándares de seguridad en los sistemas web de la Universidad Técnica de Ambato?

Esta argumentación conduce al investigador a formular las siguientes interrogantes científicas como parte de la investigación:

- ¿La revisión teórica permite conocer métodos y técnicas para el análisis de vulnerabilidades en sitios web?
- ¿La metodología OWASP tiene mecanismos que pueden conducir a resolver los problemas de vulnerabilidades en las aplicaciones web en la Universidad Técnica de Ambato?
- ¿La implementación de métodos, técnicas y herramientas seguras mediante el uso de la metodología OWASP, permitirá solucionar la seguridad de las aplicaciones web de la Universidad Técnica de Ambato?

Para resolver el problema señalado se propone como tareas investigativas las mencionadas a continuación.

Como tarea principal se tiene; Analizar las vulnerabilidades existentes en las aplicaciones web en la Universidad Técnica de Ambato, utiliza la metodología de OWASP.

Posterior a ello se detallan las tareas a cumplir para resolver el problema mencionado:

1. Fundamentación teórica y metodológica sobre métodos y técnicas usados ante amenazas en aplicaciones web.

2. Análisis de mecanismos válidos para resolver los problemas de vulnerabilidades en las aplicaciones web en la Universidad Técnica de Ambato que ofrece la metodología OWASP.
3. Evaluación de las fases de la metodología OWASP relacionados a la seguridad de las aplicaciones web de la Universidad Técnica de Ambato.

**Metodología de la Investigación**

Para el desarrollo del presente proyecto, se aplica como modalidad de investigación la bibliográfica, el estudio está basado en información que es posible encontrar en libros técnicos, informes, artículos, los cuales facilitarán la toma de información que sea relevante para llevar a cabo una investigación precisa. Se utiliza, además, la **modalidad aplicada,** se pone en práctica los conocimientos adquiridos durante el ciclo académico en lo referente a los módulos de la Maestría en Ciberseguridad de la Pontificia Universidad Católica del Ecuador y, finalmente, se aplica la **modalidad de campo** puesto que se utiliza para conocer el manejo de los procesos de la institucionales y el manejo de las diferentes aplicaciones web que existen, detecta así, las vulnerabilidades existentes, y a su vez diseñar procesos correctivos, la investigación de campo se desarrolla en la Universidad Técnica de Ambato.

**Metodología OWASP**

Se manifiesta que la Guía de Pruebas de OWASP es una metodología para un área específica. Tiene pruebas de seguridad repetidas en varias fases, los riesgos que presenta una aplicación web son prevenidos, mitigados o minimizados a través de procesos, que se implementan en cada una de las fases desde el desarrollo hasta el mantenimiento en producción de una aplicación Web.

Por otro lado Zapata (2019) indica que OWASP es una serie de buenas prácticas y recomendaciones que buscan ser una guía de trabajo enfocada a las aplicaciones desde el clico de desarrollo de software, esta metodología provee soluciones flexibles que mejoran, estandarizan y aseguran el proceso de desarrollo de una aplicación que da prioridad a la seguridad dentro del proceso de ingeniería del Software. Es importante mencionar que las vulnerabilidades pueden estar en cualquiera de los componentes de una aplicación, donde, se incluye los sistemas operativos de los equipos que las alojan, estos también presentan fallas que pueden afectar sólo el sistema o en ocasiones las aplicaciones que corren sobre el mismo, por lo tanto, un *ethical hacking* desde cualquier metodología, buscarían vulnerabilidades en ambas partes, sistema operativo y aplicación,

con el fin de exponer y reportar todos los puntos de falla por los cuales se pueda aumentar el riesgo en la compañía y, finalmente, generar cualquier tipo de afectación (p. 9).

A continuación, se muestra las fases de pruebas de seguridad de la metodología OWASP.

- Recopilación de Información.
- Pruebas de seguridad a la configuración y despliegue.
- Pruebas de seguridad a la gestión de la identidad.
- Pruebas de seguridad al proceso de autenticación.
- Pruebas de seguridad al proceso de autorización.
- Pruebas de seguridad al proceso de gestión de sesiones.
- Pruebas de seguridad a la validación de entradas.
- Pruebas de seguridad al manejo de errores.
- Pruebas de seguridad a los mecanismos criptográficos.
- Prueba de seguridad a la lógica de negocios.
- Pruebas de seguridad del lado del cliente.

**Nivel de investigación**

De acuerdo con la naturaleza del presente estudio, se realiza una investigación descriptiva y explicativa, se realizará una investigación en base a fuentes documentales que ayudaran a encontrar información para solventar las preguntas científicas que se plasman en la presente investigación, así también, con el nivel explicativo, se conoce los métodos y guías para las búsquedas de vulnerabilidades dentro de una aplicación web de la Universidad Técnica de Ambato.

**Método**

Los métodos que son utilizados en la presente investigación son: inductivo y deductivo.

El método inductivo se aplica, se demuestra los resultados obtenidos a partir de las preguntas científicas, las cuales, se solventan en su momento; El método deductivo se aplica en la presente investigación para establecer conclusiones generales de los resultados obtenidos, mediante pruebas que se realizan a una determinada aplicación web de la Institución.

**Justificación**

En la Universidad Técnica de Ambato, aún, no se cuenta con un proceso de seguridad para el control y análisis de vulnerabilidades en las aplicaciones web para reducir ataques informáticos, que se vuelve esto por muchas ocasiones un problema en cuanto al correcto funcionamiento de los servidores que proveen información para el acceso a las diferentes aplicaciones web de la Institución, que pone en peligro la integridad de toda la información importante y provoca pérdidas de recursos.

Posterior a la realización de pruebas que ayuden a detectar las diferentes vulnerabilidades en la aplicación web de la Institución, se analiza un diseño que ayude a mitigar los diferentes ataques a los que son expuestas las aplicaciones web, utiliza la menor cantidad de recursos y tiempo para solucionar problemas oportunamente, además, cabe indicar que se realiza el análisis de vulnerabilidades a una sola aplicación web, realizar dichas pruebas que ofrece la metodología OWASP a varias aplicaciones web, se torna demasiado extenso para la presente investigación.

Todo lo dicho anteriormente es posible cumplir a través de tecnologías actuales para el aseguramiento de la información, sin embargo, la virtualización de las muchas organizaciones no solo trae beneficios, sino también, conlleva riesgos que se plasman y afectan de forma negativa al correcto funcionamiento de las aplicaciones web en cuanto a la seguridad de la información, se refiere.

**Importancia**

La importancia de esta investigación se da porque permite incluir un nuevo proceso que ayude a mitigar vulnerabilidades en las aplicaciones web mediante métodos de seguridad, con la finalidad de evitar posibles ataques informáticos que podrían resultar perjudiciales para la Institución y de la misma manera a los usuarios que utilizan las diferentes aplicaciones web. Además, esta investigación es importante, será de gran beneficio para toda la comunidad universitaria, puesto que podrán estar seguros de que la información que pase a través de dichas aplicaciones tenga un alto nivel de seguridad y de esta manera tener la seguridad que la información no sea de alguna manera interceptada por algún atacante informático.

# CAPÍTULO I. ESTADO DEL ARTE Y LA PRÁCTICA

## 1.1 Seguridad en Aplicaciones Web

La importancia de este concepto para el proyecto de investigación, se explica con el aporte de varios autores.Suarez & Luis (2020) refiere que la seguridad informática, se trata sobre la protección de información de índole personal, empresarial o gubernamental contenida no solo en la red, sino también en los dispositivos de uso diario como teléfonos celulares, tabletas, computadoras de escritorio, laptop o cualquier dispositivo digital, de amenazas que puedan poner en riesgo la información almacenada o transportada en alguno de los dispositivos antes mencionados. Una buena Ciberseguridad no solo se basa en la prevención de ataques, sino también, detección y corrección de estos, se reduce los riesgos de exposición de la información, que brinda confianza a los usuarios.

Según Campderrós Vilà (2019) indica que las aplicaciones web son la cara frontal y visual del software que proporcionan el acceso a la información y a los distintos procesos de los sistemas. La información contenida en ellas suele ser el activo de mayor valor en el contexto de seguridad de la información. Esta información se suele almacenar en soportes físicos o virtuales que comprenden la infraestructura, que se debe proteger. Este suele ser el objetivo de los atacantes, los cuales intentan vulnerar la seguridad para conseguir acceso a datos sensibles, destruirlos o hacerlos inaccesibles (p. 11).

Para Marini et al. (2019) la seguridad envuelve características en concordancia con la protección del sistema, sus aplicaciones y los recursos compartidos. Incluye la prevención de la adquisición y modificación no autorizada de información, es decir, se intenta cubrir tanto la seguridad del sistema, como la de los datos del usuario (p. 2).

En concordancia con lo expuesto anteriormente por los tres autores, se deduce que la seguridad en las aplicaciones web hoy en día, se ha convertido en una parte indispensable para mantener segura la información de un sistema web, en muchas de las organizaciones la información que contiene una aplicación web emita datos sensibles que ante un posible ataque informático lograría afectar el correcto funcionamiento de dicha organización, para ello, se diseña planes que ayuden a mitigar riesgos ante un potencial ataque informático, que se presentan en diferentes sistemas web.

### 1.1.1 Requisitos para la seguridad en aplicaciones web

Según el Sistema de Gestión de la Seguridad de la Información (SGSI), toda la información almacenada y procesada por una organización está expuesta ante amenazas de ataque(por intereses comerciales, intelectuales y/o chantaje y extorsión), error (intencionado o por negligencia), ambientales (por ej. inundación o incendio), fallo en los sistemas (de almacenamiento de datos, informáticos, redes telemáticas), entre otras y también está sujeta a vulnerabilidades que representan puntos débiles inherentes a su propio uso en el ciclo de vida.

- **Confidencialidad**: la información no se pone a disposición, ni se revela a individuos, entidades o procesos no autorizados.
- **Integridad**: mantenimiento de la exactitud y completitud de la información y sus métodos de proceso.
- **Disponibilidad**: acceso y utilización de la información y los sistemas de tratamiento de esta por parte de los individuos, entidades o procesos autorizados cuando lo requieran.

Así también, la SGSI afirma que cada organización extiende e integra en un SGSI las tres características básicas iniciales de definición de la seguridad a otras adicionales como suelen ser la autenticidad, trazabilidad, no repudio, auditabilidad, según se considere oportuno para cumplir con los requerimientos internos y/o externos aplicables en cada actividad.

En correspondencia con lo anterior, se concluye que un sistema de gestión de la información se basa en tres conceptos importantes, de los cuales, se aplica procedimientos que respalden la información y a su vez el acceso a los datos de una determinada organización, que solamente usuarios autorizados hacen uso de ello.

Según Niño Benitez et al. (2018) afirma que los requisitos de seguridad para aplicaciones web se resumen en cinco principios fundamentales que son:

- **Integridad**: garantiza que los datos no sean modificados desde su creación sin autorización y que ningún intruso pueda capturar y modificar los datos en tránsito.
- **Confidencialidad**: garantiza que la información, almacenada en el sistema informático, o transmitida por la red, solamente va a estar disponible para aquellas personas autorizadas a accederla.

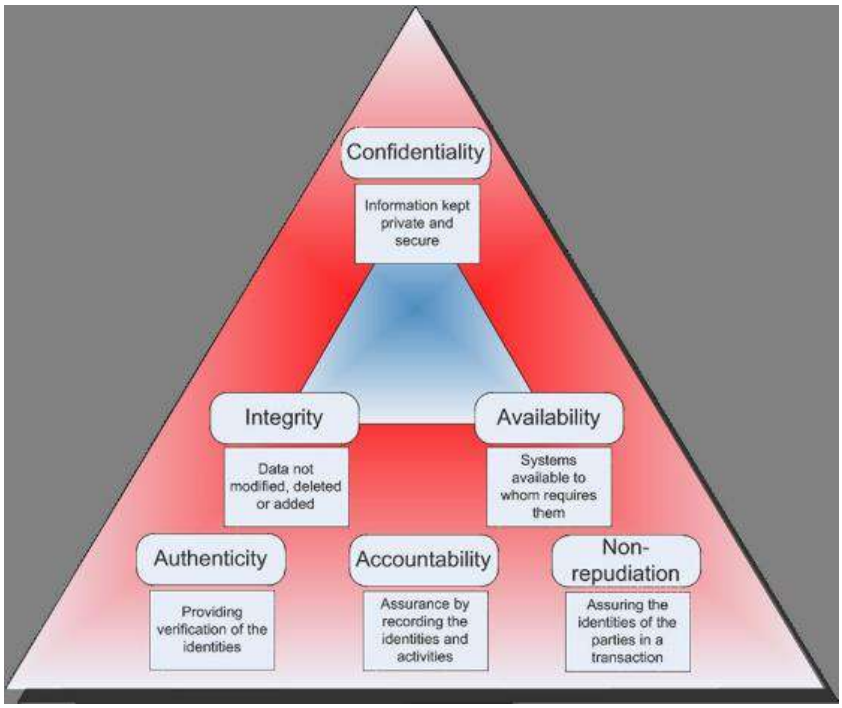
- **Disponibilidad**: garantiza el correcto funcionamiento de los sistemas de información y su disponibilidad en todo momento para los usuarios autorizados.
- **No repudio**: garantiza la participación de las partes en una comunicación. El uso y/o modificación de la información por parte de un usuario debe ser irrefutable, es decir, que el usuario no puede negar dicha acción.
- **Autenticación o Autenticidad**: asegura que sólo los individuos autorizados tengan acceso a los recursos (p. 208).

Por lo expuesto anteriormente, se deduce que se cumplirá los cinco principios fundamentales para que haya seguridad.

A continuación, se muestra en la Figura 1 los principios de la seguridad de la información en, la cual, se determina que los tres principios tienen un vínculo más estrecho, y por tal motivo son considerados como los principales en la seguridad de la información llamado la triada.

**Figura 1.**

*CIA Triad - Confidentiality, Integrity and Availability*




Fuente: *Rubaiyyaat Aakbar. 2015. Overview of Information Security for New (and non-IT) Project Managers*, Linkedin. https://www.linkedin.com/pulse/overview-information-security-new-non-it-project-managers-aakbar)

### 1.1.2 Estándares de Seguridad Informática

**ISO/IEC 27000**

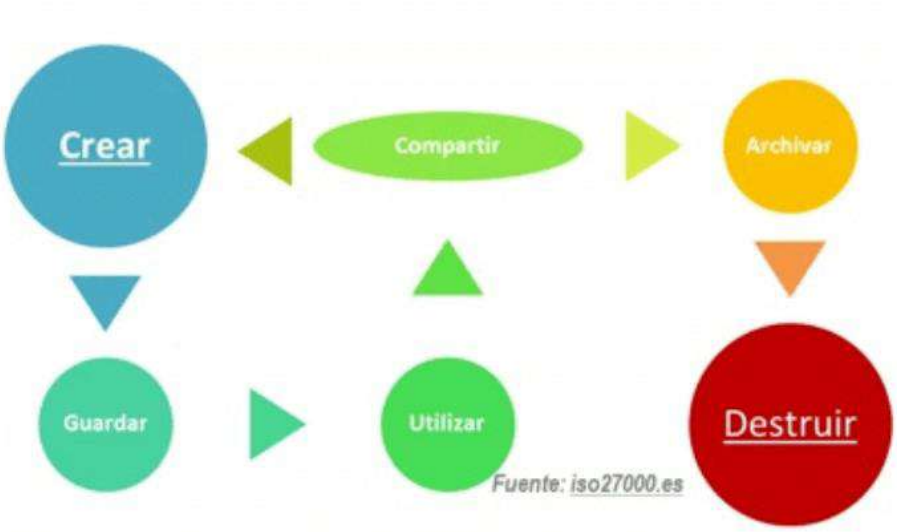
Según Torres Hallo (2020) es su artículo publicado manifiesta que se conoce que la norma ISO/ICE 27000, se conforma por un conjunto de reglamentos que han sido desarrollados por la ISO y por la IEC, con la finalidad de mejorar la gestión de la seguridad de cualquier tipo de organización, ya sea pública o privada. La importancia de la norma ISO/ICE 27000 está que sirve como una ayuda valiosa para establecer la forma de gestionar la seguridad en una organización empresarial o pública (p. 7).

Por lo dicho anteriormente, se determina que la ISO 27000 es un conjunto de estándares internacionales que dirige a la Seguridad de la Información. La ISO 27000 domina un conjunto de buenas prácticas para el manejo, implementación, mantenimiento y a su vez, la mejora de los Sistemas de Gestión de la Seguridad de la Información.

Como se ve en la Figura 2, se SGSI que presenta la ISO27000.

**Figura 2.**

*SGSI - ISO 27000*




Fuente: ISO27000. (Recuperado 11/20/2020). SGSI. https://www.iso27000.es/sgsi.html

**ISO/IEC 27001**

Reyes & Javier (2019) la definen como la norma ISO/ IEC 27001: 2013 en términos generales describe los requisitos para llevar a cabo la implementación de un Sistema de Gestión de Seguridad de la Información (SGSI), es decir, se gestiona la seguridad de la información en una empresa para, lo cual, toma como eje principal la evaluación de los riesgos, lo que permite a las organizaciones obtener una visión para definir el alcance así como el ámbito de la aplicación, normas, políticas y procedimientos a implementarse e integrar dicha metodología con el modelo de mejora continua PDCA (ciclo de Deming) frecuente en las diferentes normas de la familia ISO 27000 (p. 5).

Así mismo los datos incluidos en la Tabla 1, muestran las cifras totales de los certificados válidos y de la misma manera el número de sitios que utilizan el estándar ISO27000, durante el año 2019.

**Tabla 1.**

*Total de certificados válidos y el número total de sitios por cada estándar*

| | Total de certificados válidos | Total número de sitios |
|---|---|---|
| **ISO 9001** | 883,521 | 1,217,972 |
| **ISO 14001** | 312,580 | 487,950 |
| **ISO/IEC 27001** | 36,362 | 68,930 |
| **ISO 22000** | 33,502 | 39,651 |
| **ISO 45001** | 38,654 | 62,889 |
| **ISO 13485** | 23,045 | 31,508 |
| **ISO 50001** | 18,227 | 42,215 |
| **ISO 22301** | 1,693 | 6,231 |
| **ISO 20000-1** | 6,047 | 7,778 |
| **ISO 28000** | 1,874 | 2,403 |
| **ISO 37001** | 872 | 4,096 |
| **ISO 39001** | 864 | 1,852 |

Fuente: ISO, 2019

La norma ISO/ IEC 27001 se ha establecido como norma principal a nivel mundial para la seguridad de la información, la cual, trabaja para establecer y garantizar el aseguramiento, la confidencialidad e integridad de los datos y de la información de una organización.

### 1.1.3 Ciclo de Vida de una aplicación segura

El ciclo de vida de una aplicación web trata de asegurar que la misma, tenga los procesos y medidas adecuadas para su correcto funcionamiento, así también, se cumplirán estándares internos de la organización, con la finalidad, que se haga un uso correcto de las herramientas utilizadas para su desarrollo.

De la misma manera, se toma en cuenta que una aplicación web, se desarrolla en consideración a los objetivos y beneficios que la Institución requiera, la cual, contendrá un nivel aceptable de seguridad web, que crea un código de alta calidad y a su vez libre de vulnerabilidades.

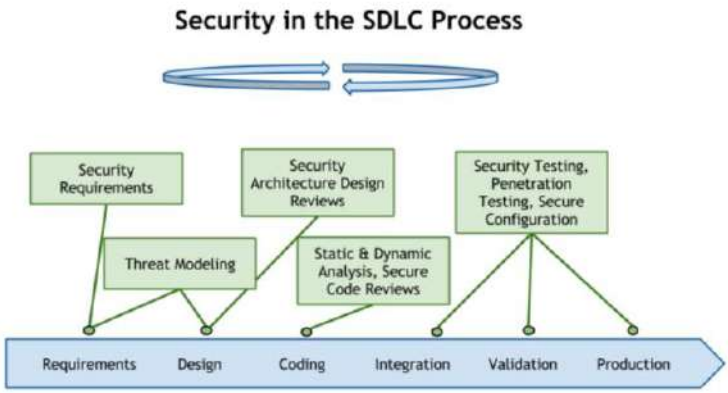
**SDLC**

Global Journals. (2020). SDLC and Development Methodologies. https://globaljournals.org/item/3918-sdlc-and-development-methodologies. Puntualiza que el SDLS (Ciclo de Vida de Desarrollo de Software en español), es una metodología cíclica, que garantiza la creación un software o aplicación, con un nivel de seguridad muy alto, se logra un software de calidad, es de gran importancia que estas contengan una capa importante de seguridad, así mismo es necesario que estas aplicaciones mantengan los principios de confidencialidad, integridad y disponibilidad y así considerarlos seguros.

Así mismo, se ve en la Figura 3, el ciclo de vida del desarrollo de software.

**Figura 3.**

*SDLC - Software Development Life Cycle*




Fuente: Ciclos de vida del Software Seguro (S-SDLC). Betabeers. Recuperado 11/30/2020.

**Fases SDLC**

Esta metodología maneja un proceso lógico para la creación o modificación de aplicaciones web, modelos y metodologías, que se usa para la creación de un software de calidad y, la cual, consta de las siguientes fases.

- Planificación
- Definición de Requisitos
- Diseño
- Desarrollo
- Pruebas
- Despliegues
- Mantenimiento

El SDLC ciclo de vida del desarrollo de software describe cada acción necesaria para crear una aplicación web. La cual, ayuda a reducir el tiempo y aumentar la eficacia del proceso de desarrollo de un software. Además, garantiza que el proyecto en sí se mantenga funcional, y a su vez desarrollado con una calidad de software adecuada.

La planificación del software llega a dividirse en investigación tecnológica, investigación de mercados y, por último, un análisis de costo-beneficio. Algunas de las tareas de las fases del SDLC se unen. Como, por ejemplo, la fase de prueba de un software se ejecuta al mismo tiempo que la fase de desarrollo de este, muchos de los desarrolladores sugieren o tienden a corregir errores que ocurren durante una etapa de prueba.

**Planificación**

En la fase de planificación, los entes involucrados en un nuevo desarrollo de software evalúan los términos que contendrá dicha creación en la cual, se toma en cuenta aspectos como el cálculo de costos, el personal, que se verá involucrado en la parte del nuevo proyecto y los materiales que incluirá, a esto se añade un cronograma, el cual, contendrá metas objetivos y los equipos a ser conformados, finalmente, un líder que tendrá el nuevo proyecto.

**Definición de Requisitos**

Rani (2017) afirma que en la fase de análisis de requisitos de SDLC donde, se discute con el cliente sobre sus necesidades con respecto al desarrollo de software. El objetivo de esta fase es capturar todos los detalles del proyecto o que la fase de análisis de requisitos; y se asegura que cada requisito y asegurarse de que todos comprendan el alcance del trabajo y cómo se cumplirá cada requisito

En conclusión, con el autor se puntualiza que la definición de requisitos es parte de la planificación, además, en esta fase, se toma en cuenta toda la información que el requirente del software tome como referencia de lo que se necesite que se automatice. Por ejemplo, si se requiere una aplicación para una entidad educativa, se toma en cuenta, los módulos que obtendrá el nuevo sistema, y toda la información de la institución para abordar de mejor manera el correcto funcionamiento de este.

**Diseño**

Rani (2017) refiere que, durante la fase de diseño, los desarrolladores y arquitectos técnicos inician el diseño de alto nivel del software y el sistema para poder cumplir con cada requisito. Los detalles técnicos del diseño se discuten con las partes interesadas y se revisan varios parámetros como los riesgos, las tecnologías, que se utilizarán, la capacidad del equipo, las limitaciones del proyecto, el tiempo y el presupuesto, y luego se selecciona el mejor enfoque de diseño para el producto.

Por lo expuesto anteriormente por el autor, se manifiesta, además, que la fase de diseño contempla detalles, como los expuestos, a continuación:

- **Arquitectura:** En, la cual, se especifica el lenguaje de programación.
- **Interfaz de Usuario:** Diseño de una plantilla que tendrá el sistema (parte visual), donde el usuario tendrá interacción con el mismo.
- **Plataformas de Ejecución:** Ya sean estas orientadas a sistemas web o escritorio.
- **Comunicaciones:** Se establece mediante la comunicación del sistema con otros servicios ya sean internos, externos u otros activos que interactúan con el sistema web.
- **Seguridad:** Define las medidas que se toma para resguardar o proteger la aplicación en, la cual, una buena práctica sería poder incluir un cifrado de tráfico SSL, también, se incluyen métodos para la protección con contraseña para accesos y almacenamiento seguro de las credenciales del usuario.

**Desarrollo**

El objetivo fundamental durante la fase de desarrollo es crear el código que contendrá los diferentes módulos de la solución y a su vez la respectiva documentación de este. El equipo de desarrollo identificará todos los requisitos a lo largo de la fase y aborda nuevas soluciones a medida que se avance con la creación del software.

Esta fase contiene tres etapas:

- Revisiones de código: La revisión determina si el código cumple con los estándares de desarrollo de software, además, sería una base fundamental en la identificación de ciertos problemas antes de la compilación, que son riesgos futuros.
- Programación en pareja: la programación en pareja si bien es cierto es una metodología de desarrollo es, la cual, se programa de una manera eficiente pero que no ahorra mucho tiempo. Pero esta técnica ayuda mucho al desarrollo del software, se mantiene una revisión continua de diseño y código, lo que conlleva a pautas de eliminación de fallas más eficientes.
- Pruebas unitarias y estáticas: mediante estas etapas, los desarrolladores validan la seguridad funcionalidad de los módulos o componentes que contendrá el nuevo software, así como verificar que los resultados que se obtiene, continuamente, logren mitigar cualquier riesgo de seguridad anteriormente identificado, a través del modelado de amenazas y el respectivo análisis de código fuente.

**Pruebas**

Según (Sharma, 2017) la fase de pruebas se realiza después de crear software para eliminar error o errores, errores para hacerlo producto de software de buena calidad sin errores. Mientras codifica por los desarrolladores y las pruebas son realizadas por expertos en pruebas en varios niveles de código, como pruebas de módulos, programas pruebas, pruebas de productos, pruebas orientadas a objetos y pruebas el producto a niveles estáticos y dinámicos. El tiempo que se dedica realiza, pruebas consume más tiempo en comparación con otras fases de SDLC.

En concordancia con el autor, se concluye que, es primordial realizar prueba de la aplicación antes de entregar el mismo a los usuarios. Generalmente, las pruebas se automatizan mediante el uso de algún otro tipo de software, así como las pruebas de seguridad. Realizar otros tipos de pruebas, se generan en ambientes específicos, es decir, se crea un lugar destino para ser o simular un ambiente

de producción, la cual, contiene los mismos procesos, que se llevaran a cabo durante el uso del software mediante usuarios finales. Las pruebas son el ente para garantizar que cada tarea dentro del sistema funcione correctamente. También, se probará los diferentes módulos de la aplicación para, que se ejecuten sin problemas, como son: prueba de rendimiento, las cuales, se utiliza para reducir los bloqueos o retrasos en el procesamiento. La fase de prueba, también, ayuda a reducir la cantidad de fallas y errores que encuentran los usuarios finales del sistema. Esto lleva a un mayor agrado del usuario final y una mejor tasa de uso.

**Despliegues/Deployment**

Según el artículo Fernando Conislla. (2020). Cómo Aplicar Seguridad En El Ciclo de Vida Del Desarrollo de Software. Recuperado de https://www.belatrixsf.com/blog/seguridad-desarrollo-software se menciona que el software debe ser desplegado en un entorno e infraestructura segura y alineada a una línea base de seguridad, se sigue políticas de *hardening* o endurecimiento personalizadas. Para completar esta etapa, deben llevarse a cabo pruebas de intrusión o *ethical hacking* independientes ejecutadas por un servicio de terceros para asegurar una revisión libre de vicios.

En concordancia con lo dicho por el autor, se menciona que esta etapa, está relacionada o está vinculada, a la seguridad de la una aplicación web, la cual, tendría una capa de seguridad, que ayudara a mejorar el rendimiento del software, y así evitar un ataque en el cual, se tome el control de una maquina y de esta manera comprometer la información de un sistema, se toma en cuenta los parámetros durante esta fase, para ello, se tomaran herramientas que ayuden a resolver cualquier brecha de seguridad, que se encuentre en un determinado software.

**Mantenimiento**

Esta fase es de gran importancia, luego de la implementación del software, el mismo contendría fallas u ocurriría cualquier error durante el funcionamiento o al agregar nuevas funciones al software. El principal objetivo del mantenimiento del software trata de eliminar errores, con la finalidad de que cada módulo o tarea presente un correcto funcionamiento, y así satisfacer al usuario final, se toma en cuenta los parámetros, que se establecieron al inicio del desarrollo del software. Existen correctivos, mantenimiento perfectivo y adaptativo para corregir dichos problemas, agregar o cambiar el funcionamiento del software sea de forma o de código.

**Modelos SDLC**

Existen varios modelos de ciclo de vida de desarrollo de software seguros, que se encuentran definidos y diseñados y que se siguen durante el proceso de desarrollo de un sistema. Estos modelos también se designan "Modelos de proceso de desarrollo de software". Cada uno de estos modelos de proceso sigue una serie de guías o pasos únicos, para garantizar la calidad y el éxito en el proceso de desarrollo de software.

A continuación, se visualizan los modelos para el SDLC más importantes que se manejan para el desarrollo correcto y adecuado para de un determinado software:

- Modelo de cascada
- Modelo iterativo
- Modelo espiral
- Modelo V
- Modelo Big Bang

## 1.2 Pruebas de Penetración

Con respecto a la información que se tiene acerca de las pruebas de penetración, se interpreta, a continuación, los siguientes conceptos en la Tabla 2.

**Tabla 2.**

*Pruebas de Penetración*

| Referencias: | Conceptos: |
|---|---|
| Vasquez & Arturo (2019) | Pruebas de penetración o Penetration Testing (Pentest) en inglés constituyen una herramienta válida para evaluar o auditar sistemas; consisten en la realización de prácticas que ponen a prueba un sistema informático, red o aplicación web para encontrar vulnerabilidades que un posible atacante podría explotar (p. 1). |
| Zafra & Luis (2017) | Una prueba de penetración o pentest es un ataque simulado y autorizado contra un sistema informático con el objetivo de evaluar la seguridad del sistema. Durante la prueba, se identifican |

| | |
|---|---|
| | las vulnerabilidades presentes en el sistema y se explotan tal como haría un atacante con fines maliciosos. Esto permite al pentester realizar una evaluación de riesgos en la actividad comercial del cliente, que se basa en los resultados de la prueba y sugerir un plan de medidas correctivas (p. 5). |
| Pozos & Inés (2019) | Las pruebas de penetración o *pentesting*, de acuerdo con la definición del libro *Penetration Testing, A Hands-On Introduction to Hacking* (Weidman, 2014), es una práctica para poder poner a prueba un sistema informático, una red o aplicación web para poder encontrar vulnerabilidades que un atacante podría explotar (p. 3). |

Fuente: elaboración propia

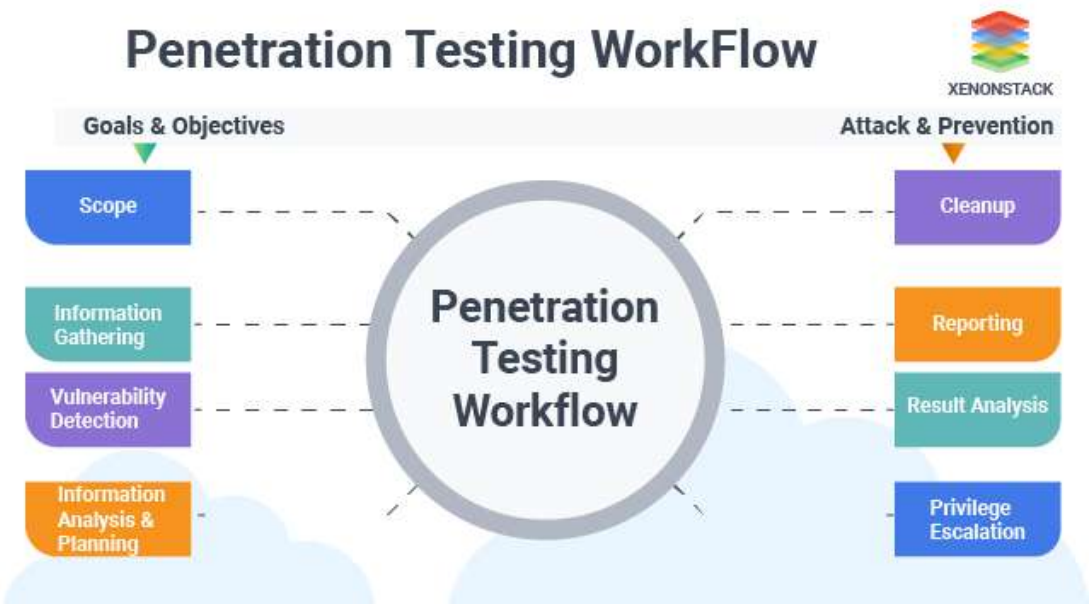
De acuerdo con los tres autores en relación con la definición de Pruebas de Penetración, se manifiesta que sería un proceso en el cual, se realizan distintos tipos de tareas o acciones a un determinado sitio o aplicación web, que se identifican, en una infraestructura objetivo, las vulnerabilidades o amenazas que podrían ser explotadas y los daños que podría causar un atacante al momento de tomar el control de un sistema informático, se toma el control de ciertos sistemas que contienen importante información dentro de una organización. Es decir, se realiza un proceso de hacking ético para identificar qué sucesos podrían ocurrir antes de que acontezcan y, posteriormente, solucionar o mejorar el sistema web, de tal manera, que se eviten estos ataques.

Otra definición de las Pruebas de Penetración está relacionada con la práctica de probar un sistema o aplicación web, red o una infraestructura para poder hallar vulnerabilidades o amenazas que un atacante podría explotar, lo cual, simula así un ataque contra los activos Tecnológicos de una organización.

A continuación, en la Figura 4, se muestra el flujo de trabajo de pruebas de penetración, que se basa en metas - objetivos y el respectivo ataque – prevención.

**Figura 4.**

*Penetration Testing WorkFlow*




Fuente: XenonStack. (2018)

### 1.2.1 Tipos de Pruebas de Penetración

DragonJar. (2003). Pruebas de Penetración. https://www.dragonjar.org/pruebas-de-penetracion.xhtml indica que existen tres tipos de Pruebas de Penetración:

- **Pruebas de Penetración de Caja Negra:** donde los pentesters o analistas de seguridad no tienen conocimiento del funcionamiento interno del sistema, y trabaja con la información que puede conseguir por sus propios medios, igual que lo podría hacer un delincuente informático.
- **Pruebas de Penetración de Caja Blanca:** en este tipo de pruebas los pentesters o analistas de seguridad tienen total conocimiento del funcionamiento interno del sistema, y trabaja con información que puede tener acceso uno o varios empleados dentro de la organización.
- **Pruebas de Penetración de Caja Gris:** donde los pentesters o analistas de seguridad pueden tener conocimiento sobre algunos aspectos del funcionamiento del sistema y de otros no.

INCIBE. (2019). INCIBE. ¿Qué es el pentesting? Auditando la seguridad de tus sistemas. https://www.incibe.es/protege-tu-empresa/blog/el-pentesting-auditando-seguridad-tus-sistemas afirma que los tipos de Pruebas de Penetración son:

- **de caja blanca**
- **de caja gris**
- **de caja negra**

### 1.2.2 Pruebas de Penetración

Según OSTEC (2018) divide los tipos de pruebas de penetración en los siguientes:

- **Prueba en Servicios de Red:** se realizan análisis en la infraestructura de red de la corporación, en busca de fragilidades que pueden ser solidificadas. En este aspecto, se evalúa la configuración del firewall, pruebas de filtrado *stateful*, etc.
- **Prueba en Aplicación Web:** es un buceo profundo en la prueba de intrusión, pues todo el análisis es extremadamente detallado y vulnerabilidades son más fácilmente descubiertas por basarse en la búsqueda en aplicaciones web.
- **Prueba de Client Side:** en este tipo de prueba, es posible explorar software, programas de creación de contenido y Web browsers (como Chrome, Firefox, Explorer y Opera entre otros) en ordenadores de los usuarios.
- **Prueba en Red Inalámbrica:** examina todas las redes inalámbricas utilizadas en una corporación, así como el propio nombre afirma. Se realiza pruebas en protocolos de red inalámbrica, puntos de acceso y credenciales administrativas.
- **Prueba de Ingeniería Social:** informaciones y datos confidenciales son pasibles de robo por medio de manipulación psicológica, un intento de inducir al colaborador a repasar ítems que deben ser sigilosos.

En concordancia con los autores se deduce que existen varios tipos de Pruebas de Penetración, las cuales fueron detalladas anteriormente como, por ejemplo, las pruebas de Caja Negra: este tipo de prueba proporciona poca o ninguna información sobre el objetivo al cual, se lo va a realizar, es decir, una aplicación web, este tipo de pruebas son más frecuentes a redes de telecomunicaciones. Por otro lado, se tienen las pruebas de Caja Blanca: son realizadas generalmente a través del personal interno de una organización, pero hoy en día es más frecuente asignarlo a un equipo externo,

quienes generalmente están asignados a un grupo de gestión de calidad de una organización, y como tal, se convierte en una parte importante dentro del ciclo de vida del desarrollo de software. Estas pruebas se realizan con el código fuente de una aplicación web, para que esta sea revisada y analizada para detectar si se encontraron o no algún tipo de debilidades o vulnerabilidades. Y, finalmente, están las pruebas de Caja Gris: las cuales requieren analizar una determinada aplicación web para obtener información importante. Este tipo de pruebas es crítico en relación con la comunicación entre el equipo de pruebas y la organización, que se encarga de la evaluación.

### 1.2.3 Métodos de Pruebas de Penetración

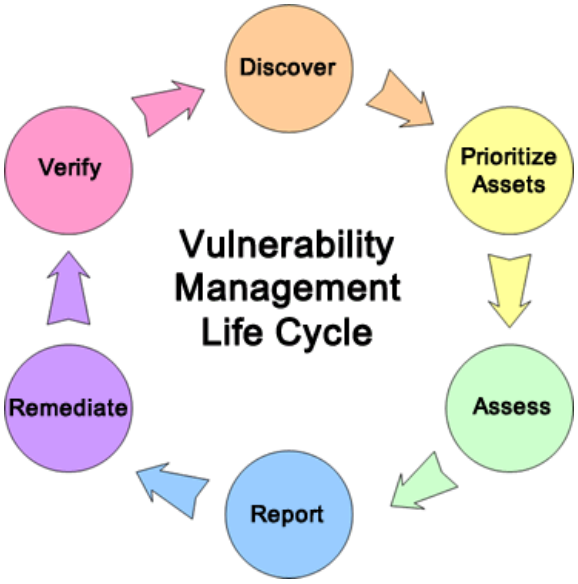
- **Manual:** es la primera manera utilizada para comprender un ataque. Aunque es manual se utilizan scripts sencillos y herramientas. Es relativamente lento en comparación con otros métodos.
- **Automático:** las herramientas se escanean rápidamente un sitio y devolver las vulnerabilidades encontradas. El inconveniente es tener menor control sobre el comportamiento del ataque, por lo cual, se es propenso a obtener falsos positivos.
- **Híbrido:** genera mejores resultados el utilizar métodos manuales y automáticos. Se utiliza un escáner de vulnerabilidades para tener una línea base y punto de inicio. Y al mismo tiempo se realizan revisiones manuales por problemas en el sitio. Los resultados del escáner serían validados, con la intención de utilizarlos luego para expandir el punto de apoyo dentro de la aplicación. Quezada. Recuperado 11/30/2020. Pruebas de Penetración contra Aplicaciones Web. Recuperado de http://www.reydes.com.

### 1.2.4 Fases Pruebas de Penetración

A continuación, en la Figura 5, se observa el ciclo de vida del análisis de las vulnerabilidades en aplicaciones Web.

**Figura 5.**

*Vulnerability Management Life Cycle*




Fuente: Vulnerability Management Life Cycle | NPCR | CDC, (2019)

En la Tabla 3 se muestra las fases para la ejecución Pruebas de Penetración a sitios web.

**Tabla 3.**

*Fases Pruebas de Penetración*

| **Referencias:** | **Conceptos:** |
|---|---|
| Romero & Yucenid (2019) | Descubrimiento y Enumeración. Es una de las etapas más importantes en un pentesting, es donde recopilamos toda la información necesaria sobre el objetivo.<br><br>Análisis de Vulnerabilidades: El análisis de vulnerabilidades es el proceso de descubrir fallas en sistemas y aplicaciones que pueden ser aprovechadas por los atacantes<br><br>Explotación: Esta es la etapa donde se realiza por parte del pentest la explotación de alguna de las vulnerabilidades encontradas en el punto anterior, aquí se saca provecho de la vulnerabilidad para intentar comprometer el sistema o aplicaciones (p. 3). |

| | |
|---|---|
| | Informe: En esta fase se hace toda la documentación para la presentación de un informe con los resultados obtenidos durante las diferentes fases ejecutadas, su finalidad es darle una visibilidad completa en donde se detallan los riesgos de todas las vulnerabilidades encontradas (p. 7). |
| Molina & Pilar (2019) | Fase de reconocimiento: En esta fase se hace uso de herramientas de análisis para obtener toda la información del objetivo (p. 24).<br><br>Fase de enumeración: En esta fase solo se realiza actividades de investigación, aun no se lleva a cabo ningún ataque (p. 24).<br><br>Fase de análisis: En esta fase se da inicio a la interacción y análisis con los sistemas encontrados, El análisis se realiza para encontrar vulnerabilidades, consultadas a nivel de la infraestructura, los sistemas operativos, los servicios disponibles o las 25 aplicaciones existentes (p. 24).<br><br>Fase de Explotación: En esta fase se inicia la intrusión en el sistema para obtener evidencia de las actividades y pasos ejecutados en las pruebas de intrusión realizada (p. 25).<br><br>Fase de Documentación: Las actividades de esta fase corresponden a realizar la documentación correspondiente al ataque realizado (p. 25). |

Fuente: elaboración propia

Como señalan los autores, se concluye que las fases de una Prueba de Penetración son:

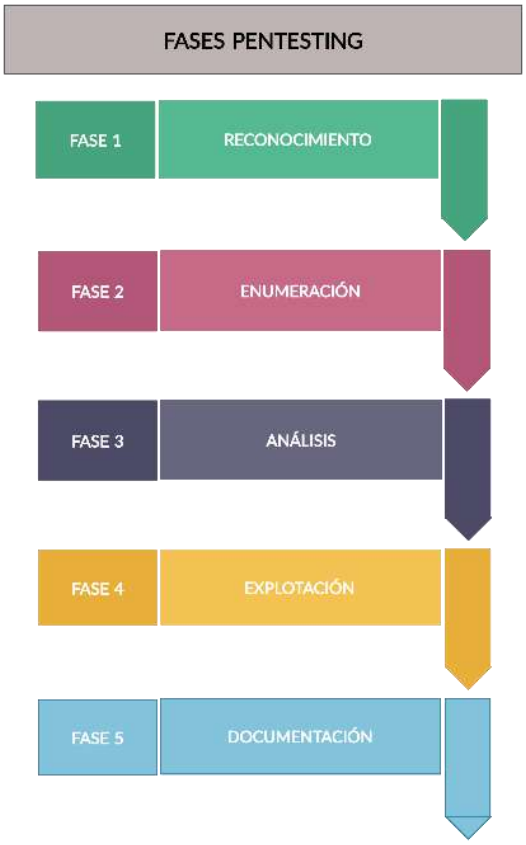
- **Reconocimiento:** es donde se proporciona los elementos para un ataque exitoso y eficiente. Además, trata de identificar el objetivo mediante varios recursos que sean de gran utilidad.
- **Enumeración:** se realiza la investigación de toda posible información que se creyere importante, para ser utilizada posteriormente.
- **Análisis:** en esta fase es donde se inicia la exploración más a fondo de la aplicación, y se determina si existen o no vulnerabilidades o alguna información para el ataque.

- **Explotación:** se toma cualquier tipo de información obtenida hasta esta fase, y se lo utiliza para explotar la aplicación. Es aquí donde se lanzan o se hacen efectivos los ataques.
- **Documentación:** en esta fase se indica que se elabora uno o varios informes en los cuales se indican las vulnerabilidades que se han encontrado y la manera de cómo se han explotado.

En la Figura 6, se muestra las fases que contempla las pruebas de penetración.

**Figura 6.**

*Fases Pentesting*




Fuente: Elaboración Propia

### 1.2.5 Metodologías para Pruebas de Penetración

En la Tabla 4 se muestra las Metodologías para las diferentes Pruebas de Penetración que se realiza a sitios web.

**Tabla 4.**

*Metodologías para Pruebas de Penetración*

| Metodología | Definición | Referencia |
|---|---|---|
| **OSSTMM (Open-Source Security Testing Methodology Manual)** | Metodología que propone un proceso de evaluación de una serie de áreas que refleja de manera fiel los niveles de seguridad presentes en la infraestructura que va a ser auditada, a estos niveles de seguridad se le denominan comúnmente "Dimensiones de Seguridad". | Romero & Yucenid (2019) |
| | Es una metodología abierta, pública y constantemente actualizada, que permite realizar la valoración de evaluación de riesgo, se obtiene una valoración cuantificable de los resultados de las pruebas realizadas. | Intriago & Karina (2018) |
| | Esta es una metodología para probar la seguridad operativa de ubicaciones físicas, interacciones humanas y todas formas de comunicaciones como inalámbricas, cableadas, analógicas y digitales. | RESEARCH, Recuperado 11/20/2020. OSSTMM. ISECOM. https://www.isecom.org/research.html |
| **ISSAF (Information Systems Security Assessment Framework)** | Marco metodológico de trabajo desarrollado por la OISSG que permite clasificar la información de la evaluación de seguridad en diversos dominios que usa diferentes criterios de prueba. | Romero & Yucenid (2019) |

| | | |
|---|---|---|
| | Identifica y evalúa las dependencias comerciales en servicios de infraestructura proporcionados por TI. Luego, llevar a cabo evaluaciones de vulnerabilidad y pruebas de penetración para resaltar las vulnerabilidades del sistema que podrían generar riesgos potenciales para los activos de información. | Intriago & Karina (2018) |
| | La metodología de prueba de penetración está diseñada para evaluar la red, controles del sistema y de la aplicación. Consiste en un enfoque de tres fases y nueve pasos evaluación. | OISSG. Recuperado 11/20/2020. Open Information Systems Security Group. https://www.oissg.org |
| **OWASP (Open Web Application Security Project)** | Metodología de pruebas enfocada en la seguridad de aplicaciones, El marco de trabajo descrito en este documento pretende alentar a las personas a evaluar y tomar una medida de la seguridad a través de todo el proceso de desarrollo | Romero & Yucenid (2019) |
| | La fundación OWASP es una organización benéfica sin fines de lucro organización con el propósito de producir herramientas, documentos, foros y capítulos gratuitos en el campo de la seguridad de aplicaciones web. La lista OWASP Top 10 se actualiza periódicamente | Willberg (2019) |

| | | |
|---|---|---|
| | documento sobre los riesgos de seguridad más críticos para las aplicaciones web. El Top 10 de OWASP 2017 se basa principalmente en más de 40 envíos de datos de empresas de seguridad de aplicaciones y una encuesta de la industria | |
| | Es una comunidad abierta dedicada a permitir que las organizaciones conciban, desarrollen, adquieran, operen y mantengan aplicaciones en las que se pueda confiar. | OWASP. Recuperado 11/20/2020. OWASP Foundation \| Open-Source Foundation for Application Security https://owasp.org. |

Fuente: elaboración propia

Por otro lado, existen varias metodologías de pruebas de penetración enfocadas a la seguridad de aplicaciones web, las cuales aseguran la obtención de los objetivos que se plantean, a fin de conocer las diferentes vulnerabilidades de un sistema informático. Dichas metodologías comprenden un conjunto de tareas y procesos que hacen que un hacker ético siga o tome en cuenta esta metodología, que asegura así la seguridad de los sistemas informáticos dentro de una organización.

## 1.3 Vulnerabilidades en Aplicaciones Web

Las aplicaciones web presentan una gran variedad de vulnerabilidades, las mismas que son evidenciadas de acuerdo con el tipo de servicio que presentan, según el sitio web SECURI, las principales vulnerabilidades que presenta el TOP 10 de la OWASP, para las diferentes aplicaciones web para el presente año son los siguientes:

A continuación, en la Figura 7, se muestra el Top 10 de las principales vulnerabilidades a sitios web según OWASP.

**Figura 7.**

*Top 10 OWASP – Web Application Security Risks*


Fuente: (SecurityTrails. Recuperado 11/30/2020. What Is OWASP?)

Otro sitio web Javier Jiménez. (27 de enero, 2020). Ataques de Inyección SQL: así afectan la seguridad. RedesZone. https://www.redeszone.net/tutoriales/seguridad/ataques-inyeccion-sql-seguridad/. publica las vulnerabilidades más comunes, las cuales son:

- Redirección a sitios maliciosos
- Recopilación de Datos
- Ataque a bases de datos
- Autenticación fraudulenta
- Ataque DDoS

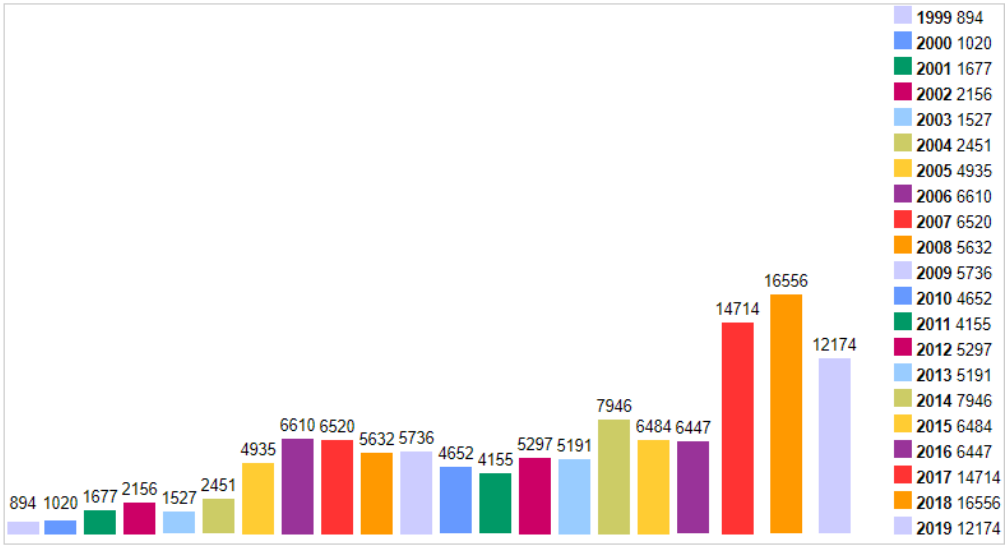
Según la compañía Positive Technologies de seguridad digital, en su publicación *Positive Technologies: 82 Percent of Web Application Vulnerabilities are in the Source Code* publicado en Febrero del 2020, manifiesta que "*El alto porcentaje de errores en el código fuente sugiere que el código fuente no está siendo revisado en busca de vulnerabilidades durante el desarrollo, lo que indica que los desarrolladores le dan un pequeño ahorro a la seguridad, en cambio, se enfocan en la funcionalidad de la aplicación*", de lo cual, se puede concluir que el principal objetivo de los ciberdelincuentes, es buscar vulnerabilidades en aplicaciones web que se desarrolla bajo código

abierto *Open Source*, para su producción, y que a su vez, puede verse limitado en cuanto a soporte técnico se refiere, lo cual es, una brecha muy importante a considerar.

Así también, se visualiza, a continuación, en la Figura 8 la evolución de las vulnerabilidades por año, las cuales son presentadas hasta el año 2019.

**Figura 8.**

*Cantidad de vulnerabilidades por año según cvedetails*



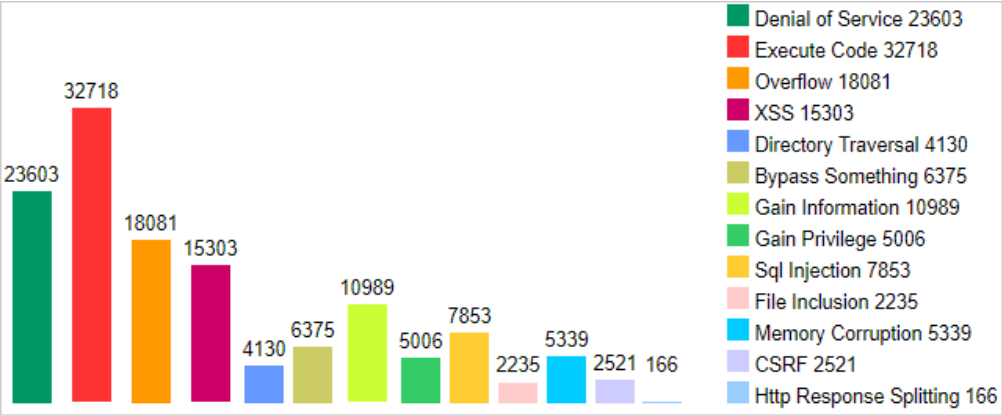

Fuente: cvedetails (2020)

De la imagen anterior se toma el análisis que las vulnerabilidades van en un aumento significativo, lo que quiere decir que durante el 2018 las vulnerabilidades fueron 16556 y que para el año 2019, si bien es cierto aún no se tiene un valor exacto del año 2019, se denota claramente un aumento considerable del mismo.

Según el sitio web cvedetails, en acorde con las vulnerabilidades por tipo, en las cuales, se demuestra que los ataque a las diferentes aplicaciones web, van cada vez en aumento como se ve, a continuación, en la Figura 9.

**Figura 9.**

*Cantidad de vulnerabilidades por tipo según cvedetails*




Fuente: cvedetails (2020)

### 1.3.1 Inyección

El Mahjoubi (2019) enuncia que la inyección de comandos es uno de los ataques más comunes en aplicaciones web en, el cual, el atacante explota alguna vulnerabilidad del sistema para ejecutar comandos SQL, NoSQL, OS o LDAP dañinos contra la voluntad de los diseñadores del sistema con el fin de acceder a datos de forma no autorizada (p. 10).

Por otro lado Chavarria Gonzalez (2020), menciona que un ataque de inyección consiste en insertar datos a una aplicación los cuales alterarán el sentido de los comandos que posteriormente serán interpretados por la aplicación (p. 60).

En concordancia con los autores se manifiesta que un ataque de inyección SQL son provocados cuando se envían datos que no son de total confianza a un intérprete de comandos (JavaScript) o similares, como parte de un conjunto de comandos o consultas. Los datos de intrusión del atacante engañan al intérprete para que este realice la ejecución comandos que no son intencionados o acceder a su vez datos sin ninguna autorización.

### 1.3.2 Autenticación Comprometida

Mahjoubi (2019) refiere que si los sistemas de autenticación y control de sesiones están implementados incorrectamente se podría dar la situación en, la cual, los atacantes mediante

diccionarios de contraseñas ejecuten un ataque de fuerza bruta para obtener acceso a un recurso o recursos de una aplicación.

Willberg (2019) afirma que las funciones de autenticación y gestión de sesiones en aplicaciones web se utilizan para verificar la identidad del usuario. La implementación incorrecta de estas funciones permite a los atacantes comprometer contraseñas, claves o tokens de sesiones. Para la creación de sesiones web, se utiliza para p.ej. mantener la preferencia de idioma del usuario.

Como señalan los autores las funciones, de las aplicaciones que se ven relacionadas con la autenticación y la gestión de sesiones comúnmente, se implementan de forma incorrecta, lo que permite a los ciberdelincuentes tomar el control de información como, contraseñas, claves o tokens de sesión de usuarios, o a su vez explotar otros tipos de fallos de implementación para asumir los roles de otros usuarios y así permanecer de forma temporal de forma permanente.

### 1.3.3 Exposición de datos confidenciales

Según Willberg (2019) la exposición de datos sensibles ocurre cuando la información no está protegida adecuadamente. Los datos sensibles incluyen, por ejemplo, información de pago, credenciales, números de teléfono, direcciones de correo electrónico, datos personales y registros de salud. Este tipo de filtraciones de datos a menudo se destructivo para las empresas.

Chavarria Gonzalez (2020) argumenta que la exposición de datos sensibles pertenece a múltiples ámbitos de una aplicación web. Desde la falta de seguridad en el canal por donde se transmiten los datos de la aplicación, hasta cómo estos se almacenan en el servidor.

Como lo hacen notar los autores, muchas aplicaciones web no realizan o administran una protección correctamente los datos confidenciales que manejan para autenticación de estas, así también, se ven afectados tanto datos de personales de usuarios, como datos financieros y de salud. Los atacantes informáticos suelen substraer o modificar dichos datos cuya protección, se ve escasa para llevar a cabo su cometido como son generalmente, fraudes o estafas con tarjetas de crédito, robos de identidad u otros delitos. Con eso se comprueba que muchos de los datos no mantienen un cifrado correcto o a su vez, no lo tienen, los datos confidenciales de un determinado usuario podrían estar en riesgo y se requiere precauciones específicas al enviarlos mediante una aplicación web, o un navegador web.

### 1.3.4 Entidades externas XML (XXE)

Campderrós Vilà (2019) refiere que un ataque de entidad externa XML es un tipo de ataque contra una aplicación que analiza la entrada XML. Este ataque ocurre cuando la entrada XML que contiene una referencia a una entidad externa es procesada por un analizador XML débilmente configurado. Este ataque puede llevar a la divulgación de datos confidenciales, denegación de servicio, falsificación de solicitudes del lado del servidor, escaneo de puertos desde la perspectiva de la máquina donde se encuentra el analizador y otros impactos del sistema (p. 83).

De acuerdo con el autor, se concluye que varios procesadores XML antiguos evalúan las referencias de entidades externas a los documentos XML. Las entidades externas son utilizadas, generalmente, para la divulgación de archivos internos, que se lo realiza mediante un gestor de archivos File Tranfer Protocol (FTP), los recursos que son compartidos de manera interna, el análisis de puertos, la ejecución de código malicioso y los ataques de DoS que no es otra cosa que la denegación de servicio.

### 1.3.5 Control de Acceso Comprometido

Willberg (2019), enfatiza que para asegurarse de que los usuarios solo puedan actuar dentro de los permisos previstos, el control de acceso se aplica en aplicaciones web. El control de acceso incluye autenticación y autorización. La autenticación es el proceso de proporcionar y validar la identidad de un usuario.

(El Mahjoubi, 2019) manifiesta que la pérdida del control de acceso puede tener un impacto muy elevado y que un atacante anónimo podría tener acceso a una cuenta de usuario administrador, hace que el atacante pueda divulgar datos sensibles o incluso modificarlos.

Las restricciones, cuando un usuario se autentica en una aplicación web lo que tienen permitido realizar generalmente no se suelen aplicar correctamente. Personas no autorizadas, se aprovechen de estas fallas para intentar vulnerar funciones o datos no autorizados que son confidenciales, y así, acceder a cuentas de otros usuarios, ver archivos confidenciales, modificar datos personales, cambiar roles de acceso, etc.

### 1.3.6 Configuración de Seguridad Incorrecto

El Mahjoubi (2019) cita que las configuraciones de seguridad incorrectas son comunes a todos los niveles del desarrollo de una aplicación. Se podrían cometer estos tipos de errores a la hora de usar *frameworks* de desarrollo con configuraciones de seguridad definidos por defecto.

Gerardo Eliasib. (2019). SPARTAN CYBERSECURITY. Configuraciones Incorrectas de Seguridad. https://hackingprofessional.github.io/Security/El-riesgo-de-las-Configuraciones-Incorrectas-de-Seguridad-OWAPS-V/ describe que una configuración errónea de seguridad surge cuando dichas configuraciones se definen, implementan y se mantienen con valores predeterminados. La buena seguridad requiere una configuración segura definida e implementada para la aplicación, el servidor web, la base de datos y la plataforma.

De lo expuesto anteriormente, se manifiesta que la configuración de seguridad incorrecta es el principal inconveniente que se observa con más periodicidad. Generalmente es el resultado de configuraciones preestablecidas inseguras, configuraciones incompletas, el almacenamiento de archivos o dispositivos en la nube sin seguridad o abierta, protocolos HTTP configurados de forma predeterminada o mensajes de error visuales, los cuales contengan información confidencial. Así mismo los sistemas operativos, páginas web, aplicaciones web y sistemas de bases de datos se configuran de forma segura y, además, preparan y actualizan frecuentemente.

### 1.3.7 Scripting entre-sitios (XSS)

Willberg (2019) manifiesta que en los ataques de Cross-Site Scripting (XSS), los scripts maliciosos se insertan en sitios web, lo que significa que son un tipo de inyección. Estos ataques pueden ocurrir en cualquier lugar La aplicación web utiliza la entrada de un usuario, sin validarla ni codificarla, en la salida que estos generan.

Campderrós Vilà (2019) define que Cross-site scripting (XSS) es un tipo de vulnerabilidad informática o agujero de seguridad típico de las aplicaciones web, que puede permitir a una tercera persona inyectar en páginas web visitadas por el usuario código JavaScript o en otro lenguaje similar (como por ejemplo VBScript) (p. 74).

Los Filtros de Scripts de Sitios (XSS) se originan cada vez que una aplicación

- Incluye datos que no son de confianza en un nuevo sitio web o aplicación sin que ésta presente algún tipo de validación o seguridad alguna.
- Actualiza un sitio web o página web que ya existe con datos que son proporcionados por el mismo usuario mediante una API de un determinado navegador que crea a su vez un código trazado mediante HTML o JavaScript. XSS permite a los ciberdelincuentes ejecutar scripts en el navegador de la víctima y substraer sesiones de usuario, dañar sitios web o también, redirigir al usuario a sitios maliciosos que contengan información desconocida.

### 1.3.8 Deserialización Insegura

GERARDO ELIASIB. (2019). SPARTAN CYBERSECURITY. Aprende Que Es Una Deserialización Insegura. https://hackingprofessional.github.io/Security/Aprende-que-es-Deserializacion-Insegura-OWASP-VII/ se indica que es una vulnerabilidad, producida cuando se usan datos no confiables para abusar de la lógica de una aplicación, infligir un ataque de denegación de servicio (DoS) o incluso ejecutar código arbitrario cuando se deserializa. Una aplicación y APIs puede ser vulnerable, si deserializan objetos hostiles o manipulados por un atacante.

Según Espíritu & Alberto (2018) indica que 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 (p. 45).

La deserialización da lugar a la ejecución de código remoto, aunque los defectos de deserialización no tienen como objetivo o resultado la ejecución de algún tipo de código remoto, se utiliza para cometer ataques como los ataques de repetición, son ataques de inyección y también ataques de escalada de privilegios de usuarios.

### 1.3.9 Uso de componentes con vulnerabilidades conocidas

Espíritu & Alberto (2018) postula que los componentes se ejecutan con privilegios asignados en la aplicación web. Si se vulnera un componente, el ataque tomará los datos o el control del servidor, por lo tanto, las vulnerabilidades de la aplicación y APIs debilitaran las defensas de las aplicaciones con diversos ataques e impactos (p. 46).

El Mahjoubi (2019) define que la vulnerabilidad o vulnerabilidades, se presenta cuando se hace uso de componentes en el desarrollo de la aplicación con componentes con vulnerabilidades ya conocidas. En este caso, conseguir un ataque exitoso sería muy fácil, sería más fácil encontrar *exploits* ya diseñados para dicha vulnerabilidad (p. 17).

Componentes como sitios web, aplicaciones web y las aplicaciones de bases de datos, se ejecutan comúnmente con los mismos privilegios de acceso. Además, los scripts que en cada aplicación web contiene, se ejecutan como recursos de esta en la cual, se mantiene total confianza en la misma, con acceso completo a todos los datos de la aplicación. Realizar un ataque basado en la generación de algún script malintencionado y puesto en marcha, se generan problemas en una determinada aplicación y a su vez encontrar puertas de acceso mediante ella a sitios no permitidos dentro de una organización. Las aplicaciones web y las APIs que usualmente utilizan algún tipo de componentes con vulnerabilidades conocidas, intentan disminuir la seguridad en ellas y así permitir varios ataques e impactos.

### 1.3.10 Registro y supervisión insuficientes

Según Espíritu & Alberto (2018) indica que es la carencia de respuesta ante ataques, que se pueden dar en el tiempo y que intenten manipular, extraer o destruir datos, hoy en día hay una brecha entre el tiempo de detección de una amenaza, es típicamente detectado por terceros en lugar de por procesos internos (p. 47).

Willberg (2019) agrega que esta no es una vulnerabilidad en sí misma, sino más bien un problema que hace posible que los atacantes para llevar a cabo un ataque. El registro y el monitoreo insuficientes son la base de casi cada incidente importante. La falta de supervisión y respuesta oportuna permite a los atacantes lograr sus objetivos sin ser detectado

Por lo mencionado por los autores, se recalca la importancia de asegurar un sitio web, este no se subestima. Si bien es cierto que el 100% de seguridad no es un objetivo principal y mucho menos realista, existen formas de mantener un sitio web seguro y también monitoreado de forma regular, para que tenga que tomar medidas prontas cuando algo suceda en dichas aplicaciones. El no tener implementado un proceso de registro y monitoreo eficiente y eficaz aumenta el deterioro de un sitio web y este se ve en ocasiones comprometido.

A continuación, en la Figura 10, se muestra el riesgo top 10 de la metodología OWASP.

**Figura 10.**

*Clasificación de Riesgos según OWASP*

| Riesgo OWASP | Probabilidad e impacto | Resultados | | | | | |
|---|---|---|---|---|---|---|---|
| | | Pruebas pretest sin OWASP | | | Pruebas postest con OWASP | | |
| A1 | Probabilidad | ALTO | 7.875 | 3.500 | BAJO | 0.910 | 1.300 |
| | Inpacto | | | 2.250 | | | 0.700 |
| A2 | Probabilidad | CRITICO | 12.160 | 4.000 | BAJO | 0.006 | 1.070 |
| | Inpacto | | | 3.040 | | | 0.006 |
| A3 | Probabilidad | ALTO | 7.343 | 3.575 | BAJO | 2.268 | 2.257 |
| | Inpacto | | | 2.054 | | | 1.005 |
| A4 | Probabilidad | ALTO | 6.241 | 2.051 | BAJO | 1.376 | 1.345 |
| | Inpacto | | | 3.043 | | | 1.023 |
| A5 | Probabilidad | MEDIO | 4.252 | 2.067 | BAJO | 1.004 | 1.003 |
| | Inpacto | | | 2.057 | | | 1.001 |
| A6 | Probabilidad | ALTO | 6.150 | 3.075 | BAJO | 2.545 | 1.777 |
| | Inpacto | | | 2.000 | | | 1.432 |
| A7 | Probabilidad | ALTO | 6.027 | 3.003 | BAJO | 1.900 | 1.324 |
| | Inpacto | | | 2.007 | | | 1.435 |
| A8 | Probabilidad | ALTO | 7.585 | 2.957 | BAJO | 2.757 | 1.650 |
| | Inpacto | | | 2.565 | | | 1.671 |
| A9 | Probabilidad | ALTO | 8.974 | 3.654 | BAJO | 1.009 | 1.004 |
| | Inpacto | | | 2.456 | | | 1.005 |
| A10 | Probabilidad | ALTO | 8.467 | 3.446 | BAJO | 2.916 | 1.763 |
| | Inpacto | | | 2.457 | | | 1.654 |

Fuente: Espíritu & Alberto (2018)

- Inyección - A1
- Autenticación Comprometida - A2
- Exposición de datos confidenciales - A3
- Entidades externas XML (XXE) - A4
- Control de Acceso Comprometido - A5
- Configuración de Seguridad Incorrecto - A6
- Filtros de Scripts de Sitios (XSS) - A7
- Deserialización Insegura - A8
- Uso de componentes con vulnerabilidades conocidas - A9
- Registro y supervisión insuficientes - A10

# CAPÍTULO II. DISEÑO METODOLÓGICO

## 2.1 Caracterización de la Institución

La Universidad Técnica de Ambato, siendo una institución pública que brinda el servicio de educación superior, fue creada el 18 de abril de 1969 según aprobación del Congreso Nacional. Nació con el lema "Educarse es aprender a ser libres" bajo el pensamiento y la égida del Doctor Carlos Toro Navas quien presidió la conformación del Primer Consejo Universitario, luego de realizada la primera Asamblea Universitaria un 10 de mayo de 1969. Vicerrector fue designado el economista Víctor Cabrera Guzmán.

El 28 de junio de 1963, en el Gobierno Constitucional de la República del Dr. Carlos Julio Arosemena Monroy, se dictó el Decreto promulgado en el Registro Oficial No. 499 de 5 de julio de 1963, por el cual, fue oficializado este Centro de Educación Superior, reconociéndole personería jurídica, autonomía y estableciendo, además, el hecho legal de que actuará y estará amparado por la Ley de Educación Superior, es decir, se considera como Instituto a nivel Superior, expide la presente ley.
Art. 1.- Créase la "Universidad Técnica de Ambato", a base del actual Instituto Superior, y que funcionará en la misma ciudad, por el momento con las facultades de Contabilidad Superior y Auditoría, de Gerencia y Administración y de Técnica Industrial, esta última con las Escuelas; a) Tecnología de Cuero y Caucho; y b) Tecnología de Alimentos; pudiendo la Universidad Técnica con su autonomía crear en el futuro nuevas Facultades, de acuerdo con las necesidades de la provincia y de la economía, con la que se cuenta y se disponga, tanto más que no existe otra Universidad de ésta índole en el País.

**Visión**

Formar profesionales líderes competentes, con visión humanista y pensamiento crítico a través de la Docencia, la Investigación y la Vinculación, que apliquen, promuevan y difundan el conocimiento respondiendo a las necesidades del país.

**Misión**

La Universidad Técnica de Ambato por sus niveles de excelencia se constituirá como un centro de formación superior con liderazgo y proyección nacional e internacional.

**Aplicaciones Web**

La Universidad Técnica de Ambato, cuenta con aplicaciones web que son en su mayoría de gran importancia, lo que quiere decir es que cada una de ellas maneja datos confidenciales para quienes utilizan las diferentes plataformas web.

Por lo dicho anteriormente, para el presente proyecto, se toma en consideración una de las aplicaciones web que maneja datos sensibles para los usuarios y a su vez sea de uso constante por los mismos, por la gran cantidad de información relevante que las aplicaciones web de la Institución maneja, se toman en consideración únicamente una aplicación web a la cual, se utiliza las diferentes pruebas de seguridad que usa la Guía de Pruebas que ofrece la metodología OWASP en su versión 4.0, dicha aplicación web, será aquella que más demanda de uso refiera.

A continuación, se muestra las aplicaciones web, disponibles dentro de la Institución.

**Tabla 5.**

*Lista de Aplicaciones Web*

| Nombre aplicación | Lenguaje de Programación | Utilidad | Tipo de Software | Enlace de acceso |
|---|---|---|---|---|
| Sistema de Bibliotecas | PHP | Manejo de Libros | Propio de la Institución | https://bibliotecas.uta.edu.ec |
| Proyectos de Investigación | PHP | Evidencias de Proyectos de Investigación | Propio de la Institución | https://servicios.uta.edu.ec |
| Practicas Preprofesionales | PHP | Evidencias de Practicas Preprofesionales | Propio de la Institución | https://servicios.uta.edu.ec |
| Silabo de Posgrado | PHP | Información sobre el silabo de los docentes | Propio de la Institución | https://servicios.uta.edu.ec |
| REDI | PHP | Repositorio de material digital de la Comunidad Universitaria | Adaptado | http://redi.uta.edu.ec |
| Planificación estrategia | PHP | Documentación sobre el plan estratégico de desarrollo institucional | Propio de la Institución | https://servicios.uta.edu.ec |

| | | | | |
|---|---|---|---|---|
| Silabo de Grado | PHP | Información sobre el silabo de los docentes | Propio de la Institución | https://servicios.uta.edu.ec |
| Registro de Asistencia Docente | C# | Registro de Asistencia de los Docentes a clases regulares | Propio de la Institución | https://controldocente.uta.edu.ec |
| Becas Estudiantiles | ASP.Net C# | Información acerca de las asignaciones de becas a los estudiantes | Propio de la Institución | https://servicios.uta.edu.ec |
| Seguimiento a estudios de Posgrado | ASP.Net C# | Evidencias de los Docentes que se encuentran cursando un Posgrado | Propio de la Institución | https://servicios.uta.edu.ec |

Fuente: elaboración propia

El presente análisis de vulnerabilidades se enfocará exclusivamente en la aplicación de Bibliotecas por lo mencionado anteriormente, cuyas características son las siguientes:

- Alojamiento: Servidor WEB dentro de la Institución
- Desarrollo propio o adquirido: Adquirido
- Lenguaje: PHP
- Año: 2018
- Base de datos: PostgreSQL – SQL Server
- Usuarios: Comunidad Universitaria
- Transacciones mes: 300
- Funcionalidades: Búsqueda de libros Institucionales

Se ejecutará el análisis y detección de vulnerabilidades que utiliza mecanismos basados en la metodología OWASP, que genera un informe de riesgos y se establecen los mecanismos aplicables al resto de aplicaciones con que cuenta la UTA, de esta manera se delimita la investigación al análisis de una solo aplicación seleccionada al azar y que sirve de referencia para un futuro análisis otras aplicaciones y servicios web que dispone la institución.

## 2.2 Metodología de Investigación

Para la presente investigación se utilizará el método Inductivo, la cual, va de lo particular a lo general. Lo que quiere decir que es parte de conocimientos básicos de un tema en particular y

hechos específicos que fueron parte de una investigación, posterior a ello, se utiliza la generalización que llega a establecer reglas y leyes científicas. Este método está basado en la experiencia, la observación y en los hechos que fueron debidamente orientados para que un estudiante logre la perseverancia en alguna práctica que estuviere desarrollando, además, se utilizara el Método Deductivo, el cual, permite presentar conceptos, reglas, principios definiciones a partir de las cuales se analiza, se sintetiza compara, generaliza y demuestra.

**Modalidad de Investigación:** La presente es una investigación de campo, la misma que permite recopilar toda la información que se requiere en el lugar de los hechos, por su naturaleza se plantea una posible solución al mejoramiento del problema en contexto previo al desarrollo de la investigación. De la misma manera, se realiza una revisión bibliográfica, la cual, permite recopilar información ordenada y documentada de revistas científicas, artículos científicos e internet, y así poder sustentar de manera teórica la investigación del presente proyecto.

Así también, se realiza una investigación experimental, la cual, centraliza en controlar el fenómeno a estudiar en, la cual, se emplea el razonamiento hipotético-deductivo, que parte de unas premisas teóricas dadas se llega a una conclusión. Además, se emplea muestras representativas, el diseño experimental cuenta como estrategia de control y metodología cuantitativa para analizar los datos.

**Tipo de Investigación:** Los tipos de investigación que se emplean en la investigación son: descriptivas y explicativas, la investigación descriptiva es la que describe los datos y características de la población o fenómeno que va a ser parte de un determinado estudio o investigación, la investigación explicativa se utiliza porque permite un análisis del fenómeno de estudio para su corrección.

**Instrumentación:** Para la presente investigación para la recolección de datos se utiliza el instrumento y técnica de la entrevista, la cual, contendrá preguntas que se enfocan a la seguridad dentro de las aplicaciones web, la misma que se encuentra desarrollada en el Anexo 1, además, aplica la técnica de la observación, por medio de ella, se logra determinar el desarrollo y el de las actividades dentro de la institución, si existe la aplicación de barreras y, así también, procedimientos de control, como medidas de prevención y contramedidas ante algún tipo de amenaza a los recursos informáticos y la respectiva confidencialidad de la información. Se realiza la observación al

departamento de tecnología de la Institución para obtener información relacionada a la seguridad en las aplicaciones web de la Institución. Este proceso se realizó mediante la aplicación de una ficha de observación que se encuentra en el Anexo 2, para llevar constancia de lo observado durante el segundo semestre del presente año.

**Herramientas de Software:** A continuación, se enuncia algunas de las herramientas a ser utilizadas para la etapa de pruebas a ser aplicadas a las diferentes aplicaciones web de la Institución.

**Kali Linux:** Kali Linux es un proyecto de código abierto mantenido y financiado por Offensive Security, un proveedor de servicios de prueba de penetración y capacitación en seguridad de la información de clase mundial. KALI. Recuperado 11/30/2020. KALI. Our Most Advanced Penetration Testing Distribution, Ever. https://www.kali.org.

**OWASP ZAP:** El *Zed Attack Proxy* (ZAP) es una herramienta de pruebas de penetración integrada, fácil de usar para encontrar vulnerabilidades en aplicaciones web. Está diseñada para ser utilizada por personas con amplia experiencia en seguridad y, como tal, es ideal para desarrolladores y evaluadores funcionales que son nuevos en el uso de pruebas de penetración. ZAP. Recuperado 11/30/2020. The ZAP Homepage. https://www.zaproxy.org.

**Nikto:** Es un escáner de servidor web de código abierto que realiza pruebas exhaustivas contra servidores web para varios elementos, incluidos más de 6700 archivos / programas potencialmente peligrosos, verifica versiones desactualizadas de más de 1250 servidores y problemas específicos de la versión en más de 270 servidores. CIRT.net. Recuperado 11/30/2020. Nikto2. https://cirt.net/Nikto2.

**Httprint:** Es una herramienta de huellas dactilares de servidor web. Se basa en las características del servidor web para identificar con precisión los servidores web, a pesar de que pueden haber sido ofuscados al cambiar las cadenas de banner del servidor o mediante complementos como mod_security o server mask. httprint también, se usa para detectar dispositivos habilitados para la web que no tienen una cadena de banner de servidor, como puntos de acceso inalámbricos, enrutadores, conmutadores, módems de cable, etc. NET SQUARE. Recuperado 11/30/2020. Httprint. https://net-square.com/httprint.html.

**Shodan:** Es una herramienta que se utiliza para descubrir cuáles de sus dispositivos están conectados a Internet, dónde se encuentran y quién los está utilizando. Shodan. Recuperado 11/30/2020. Shodan. https://www.shodan.io.

**Gregthatcher:** Es una herramienta que incluye una amplia gama de herramientas que permiten a los administradores del sistema "ver" qué tan bien están funcionando sus hosts en Internet. Haga clic en uno de los enlaces a continuación para obtener más detalles sobre las funciones de InternetPeriscope. Gregthatcher Recuperado 11/30/2020. InternetPeriscopio. http://www.gregthatcher.com.

**Httprecon:** El proyecto httprecon investiga en el campo de las huellas dactilares del servidor web, también conocido como huellas dactilares http. El objetivo es la identificación altamente precisa de implementaciones httpd dadas. Esto es muy importante dentro del análisis de vulnerabilidad profesional. Computec. Recuperado 11/30/2020. Httprecon Project. https://www.computec.ch/projekte/httprecon/.

**Burp Suite:** Es un conjunto de herramientas manuales con funciones limitadas para explorar la seguridad web. Aproveche su tráfico HTTPS, edite y repita solicitudes, decodifique datos y más. PortSwigger**.** Recuperado 11/30/2020. PortSwigger es líder en ciberseguridad. https://portswigger.net/burp.

**DNSstuff:** Es una web en la que podemos encontrar una gran cantidad de recursos, gratuitos y de pago, para permitirnos monitorizar redes, servidores o cualquier otro recurso IP de manera que podamos conocer si todo está correcto, si hay problemas de rendimiento y, de haber algún otro tipo de problema, poder solucionarlo fácilmente. RedesZone. Recuperado 11/30/2020. DNSstuff, conoce este completo kit de herramientas gratuito para monitorizar dominios, redes, IPS y más. https://www.redeszone.net/2018/12/22/dnsstuff-herramientas-monitorizar-redes/.

**WhatWeb:** Es un escáner web de próxima generación. reconoce tecnologías web que incluyen sistemas de gestión de contenido (CMS), plataformas de blogs, paquetes de estadísticas / análisis, bibliotecas JavaScript, servidores web y dispositivos integrados. WhatWeb. Recuperado 11/30/2020. WhatWeb. https://www.whatweb.net/.

**NetCraft:** Proporciona servicios de seguridad de Internet para una gran cantidad de casos de uso, incluida la detección e interrupción de delitos cibernéticos, las pruebas de aplicaciones y el escaneo PCI. También analizamos muchos aspectos de Internet, incluida la cuota de mercado de servidores web, sistemas operativos, proveedores de alojamiento, autoridades de certificación SSL y tecnologías web. NetCraft. Recuperado 11/30/2020. NetCraft. https://sitereport.netcraft.com/.

**Wappalyzer:** Es un generador de perfiles de tecnología y proveedor líder de datos. Nuestros productos proporcionan a los equipos de ventas y marketing conocimientos y herramientas tecnológicos para la generación de leads, el análisis de mercado y la investigación de la competencia. Wappalyzer. Recuperado 11/30/2020. Wappalyzer. https://www.wappalyzer.com/.

**Screaming Frog SEO Spider:** Es un rastreador web que le ayuda a mejorar el sitio de SEO, mediante la extracción de datos y auditoría para los problemas comunes de SEO. ScreamingFrog. Recuperado 11/30/2020. Screaming Frog SEO Spider. SEO Spider Tool. https://www.screamingfrog.co.uk/seo-spider/.

**Hstspreload:** Se utiliza para enviar dominios para su inclusión en la lista de precarga HTTP Strict Transport Security (HSTS) de Chrome. Esta es una lista de sitios que están codificados en Chrome como solo HTTPS. Hstspreload. Recuperado 11/30/2020. Hstspreload. https://hstspreload.org/.

**SSL Server Test:** realiza un análisis profundo de la configuración de cualquier servidor web SSL en la Internet pública. Qualys SSL Labs. Recuperado 11/30/2020. SSL Server Test. https://www.ssllabs.com/ssltest/.

**W3af:** Es un marco de auditoría y ataque de aplicaciones web. El objetivo del proyecto es crear un marco que le ayude a proteger sus aplicaciones web mediante la búsqueda y explotación de todas las vulnerabilidades de las aplicaciones web. W3af. Recuperado 11/30/2020. W3af. http://w3af.org/.

**DotDotPwn:** Es una herramienta de fuzzing inteligente que permite a un atacante detectar vulnerabilidades potenciales que pueden estar relacionadas con el directorio transversal dentro de un servicio dado. La herramienta es efectiva y puede ayudar a descubrir fallas en protocolos de servidor web como TFTP, HTTP y FTP. La herramienta puede ser especialmente útil, cuando se manejan pruebas de penetración en aplicaciones basadas en web. CyberPunk. 2016. DotDotPwn: The Directory Traversal Fuzzer. https://www.cyberpunk.rs/dotdotpwn-the-directory-traversal-fuzzer.

**WebScarab:** Está diseñado para ser una herramienta para cualquier persona que necesite exponer el funcionamiento de una aplicación basada en HTTP (S), ya sea para permitir que el desarrollador depure problemas que de otro modo serían difíciles o para permitir que un especialista en seguridad identifique vulnerabilidades en la forma en que la aplicación ha sido diseñado o implementado. Kali

Tools. Recuperado 11/30/2020. WebScarab Package Description. https://tools.kali.org/web-applications/webscarab.

**Testssl:** Es una herramienta de línea de comandos gratuita que verifica el servicio de un servidor en cualquier puerto para el soporte de cifrados TLS / SSL, protocolos, así como fallas criptográficas recientes y más. Testssl. Recuperado 1/30/2020. Testing TLS/SSL encryption. https://testssl.sh/.

**Curl:** Es una herramienta para transferir datos desde o hacia un servidor, utiliza uno de los protocolos compatibles (DICT, FILE, FTP, FTPS, GOPHER, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, MQTT, POP3, POP3S, RTMP, RTMPS, RTSP, SCP, SFTP, SMB, SMBS, SMTP, SMTPS, TELNET y TFTP). El comando está diseñado para funcionar sin interacción del usuario. Curl. Recuperado 11/30/2020. curl.1 the man page. https://curl.se/docs/manpage.html.

**GNU Wget:** Es un paquete de software gratuito para recuperar archivos mediante HTTP, HTTPS, FTP y FTPS, los protocolos de Internet más utilizados. Es una herramienta de línea de comandos no interactiva, por lo que se puede llamar fácilmente desde scripts, terminales sin soporte para X-Windows, etc. Gnu. Recuperado 11/30/2020. GNU Wget. https://www.gnu.org/software/wget/.

**Nmap:** Es un código abierto y gratuito (licencia) utilidad para el descubrimiento de redes y la auditoría de seguridad. Nmap. Recuperado 11/30/2020. Nmap.org. https://nmap.org/.

## 2.3 Metodología OWASP

En este capítulo se describe las pruebas de seguridad, que se va a realizar aplicaciones web de Bibliotecas de la Universidad Técnica de Ambato, utiliza la Guía de Pruebas OWASP v4.0.

Guía de Pruebas: Esta guía, considerada la más interesante dentro del conjunto documental de la metodología OWASP, resume y evidencia las vulnerabilidades o fallas de seguridad en las aplicaciones y cómo un intruso explota estos puntos de entrada. La guía posee ejemplos gráficos y contundentes que están perfectamente explicados, con el fin de concientizar a las organizaciones o a quién usa la guía de que el peligro es real, de que las aplicaciones mal desarrolladas son un riesgo inminente que podría traer una serie de consecuencias negativas, que afecta la funcionalidad normal de la organización (Zapata, 2019).

- Recopilación de Información.
- Pruebas de seguridad a la configuración y la implementación.

- Pruebas de gestión de la identidad.
- Pruebas de autenticación.
- Pruebas de autorización.
- Pruebas de gestión de sesiones.
- Pruebas de validación de entradas.
- Pruebas al manejo de errores.
- Pruebas de criptografía débil.

### 2.3.1 Recopilación de Información

En esta fase lo que primero se requiere es la búsqueda de la mayor cantidad de información acerca de las aplicaciones web, la recopilación de información es un paso necesario en una prueba de intrusión.

Para identificar de mejor manera la categoría a la que pertenecen cada una de las pruebas que ofrece la metodología OWASP, se ha creado una nomenclatura por cada una de ellas, que se forma de la siguiente manera.

**OTG-INFO-001**

**OTG*:*** OWASP TEST GUIDE v4.0 (Guía de Pruebas de OWASP v4.0)

**INFO:** Se refiere a la categoría de cada fase, en este caso (Recopilación de información)

**001:** Se refiere al número de categoría a la que pertenece, para el caso es: prueba número uno de la categoría recopilación de Información.

- **Realizar el descubrimiento y reconocimiento del motor de búsqueda para la fuga de información (OTG-INFO-001)**

**Objetivo de la Prueba**

El objetivo de la prueba es buscar la mayor cantidad de información en el Internet, acerca del sitio web que se analiza y, además, verificar si existe algún tipo de información expuesta.

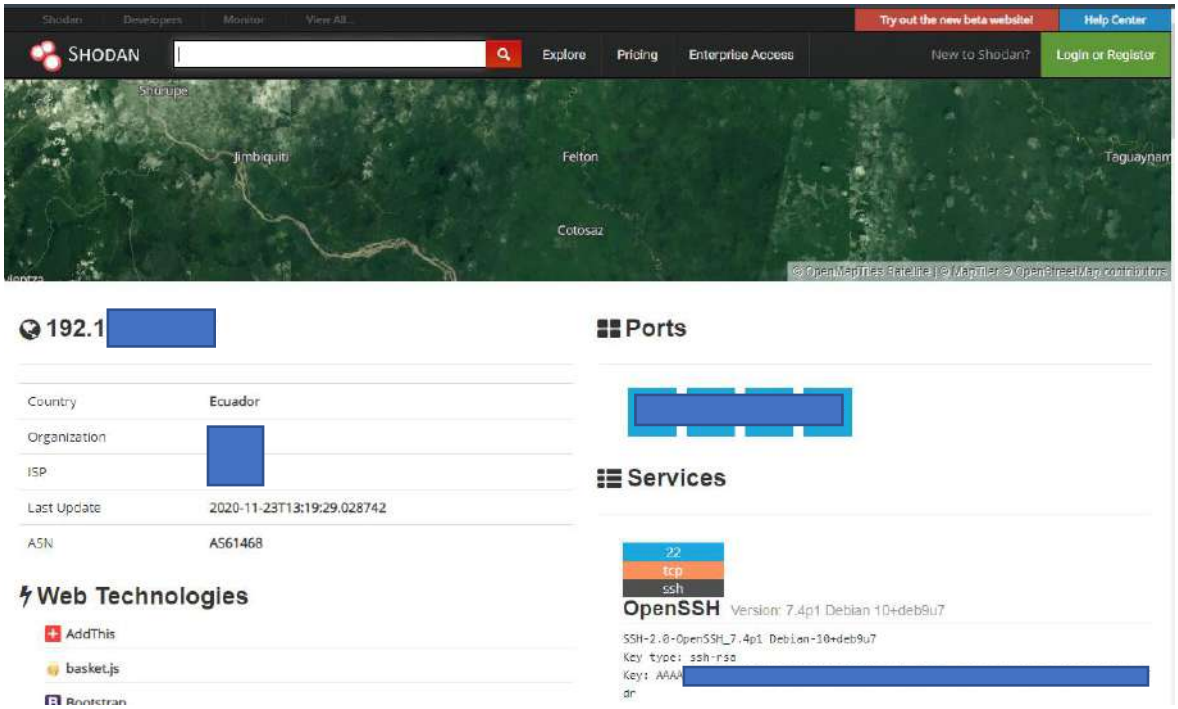
**Ejecución de la Prueba**

Para realizar la comprobación, se utilizó las herramientas: *Google, Shodan, Google Dorks*.

A continuación, en la Figura 11, se muestra la recopilación de la información de la aplicación web como, por ejemplo, la dirección IP, puertos utilizados, tecnologías web y servicios que se ejecuta.

**Figura 11.**

*Recopilación de Información – Herramienta Shodan*



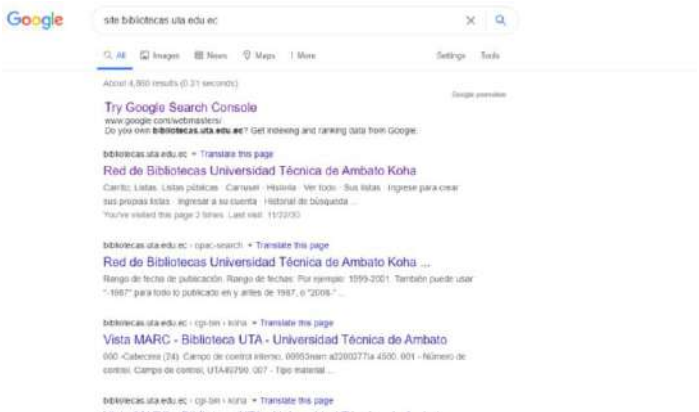

Fuente: elaboración propia

Google: Se utiliza para verificar si existen subpáginas dentro de la aplicación web

Para verificar un archivo indexado mediante Google la sintaxis es: site: ejemplo.com, como indica en la Figura 12 y, la cual, indica que existen varias páginas asociadas al sitio web de Bibliotecas.

**Figura 12.**

*Recopilación de Información – Google site*




Fuente: elaboración propia

Para mostrar la página principal de la aplicación web como esta en cache la sintaxis es: cache: ejemplo.com, como se muestra en la Figura 13, la cual, indica que si esta la página web en cache.

**Figura 13.**

*Recopilación de Información – Google cache*


Fuente: elaboración propia

- **Uso de huellas digitales en el Servidor Web (OTG-INFO-002)**

**Objetivo de la Prueba**

El objetivo de la prueba es encontrar la versión y el tipo de servidor, donde se aloja la aplicación web que es el objetivo, en las cuales, se podría encontrar vulnerabilidades conocidas y la manera de explotarlas durante las pruebas.

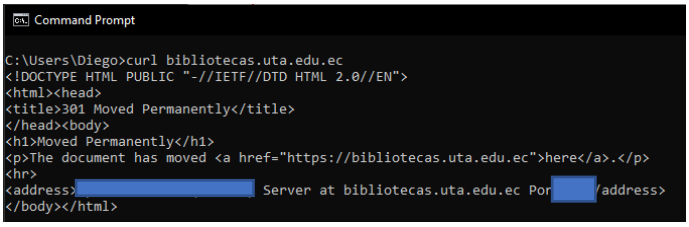
**Ejecución de la Prueba**

Para obtener información sobre el servidor web, se obtiene de la siguiente manera mediante la herramienta curl.

En la Figura 14, se muestra que mediante el comando curl, se consigue la información del tipo y la versión del servidor donde se aloja la aplicación.

**Figura 14.**

*Información del Servidor Web – Herramienta Curl*

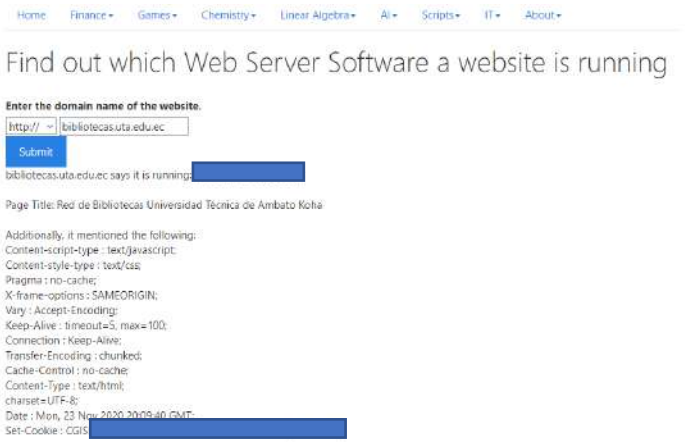



Fuente: elaboración propia

A continuación, en la Figura 15, se muestra la información del servidor donde se aloja la aplicación.

**Figura 15.**

*Información del Servidor Web – Herramienta Gregthatcher*

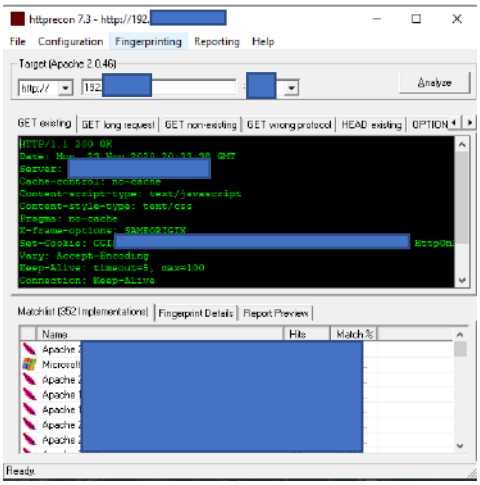



Fuente: elaboración propia

A continuación, en la Figura 16, se muestra el análisis mediante pruebas automatizadas la información del servidor, tipo y versión del servidor web.

**Figura 16.**

*Información del Servidor Web – Herramienta Httprecon*




Fuente: elaboración propia

- **Revisión de Meta-archivos del servidor web en busca de fugas de información (OTG-INFO-003)**

**Objetivo de la Prueba**

El objetivo de la prueba es buscar fugas de información de la ruta o rutas al directorio o carpeta de la aplicación web, que se lo realiza mediante el uso de la información del archivo robots.txt. Además, este crea la lista de archivos de directorios los cuales no son *indexados* por arañas, *robots* o rastreadores.

El archivo robots.txt tiene el siguiente contenido, se muestra la información más importante:

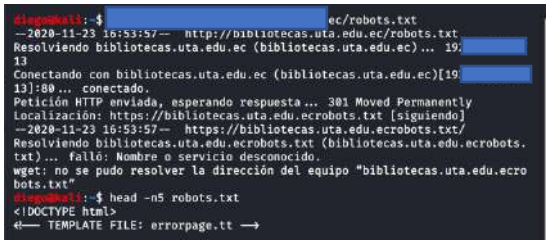
- ***User Agent: *:*** *Robots* de los motores de búsqueda, las cuales harían caso todas las reglas que se detallen, que es el caso de todos.
- ***Disallow: /templates:*** Especifica que recurso están prohibidos, niega el acceso a todos los archivos almacenado en el directorio *templates*.
- ***Allow: /images***: Permite el acceso a los archivos que se encuentran almacenados en el directorio *images*
- ***Sitemap:*** Incluye información sobre el mapa del sitio en un formato *xml*.

**Ejecución de la Prueba**

Para realizar esta prueba, se utiliza un navegador *web* y la herramienta wget, con la finalidad de encontrar archivos que contengan información de importancia para un posible ataque. En la Figura 17, se observa el contenido del archivo robots.txt enlazado a la aplicación web, pero para el caso no se muestra la información, además, no se tiene acceso a la misma.

**Figura 17.**

*Información robots.txt – Herramienta Wget*

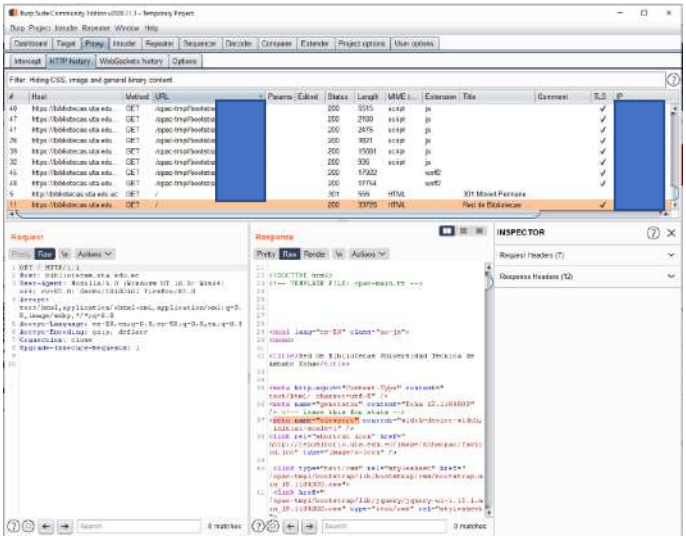



Fuente: elaboración propia

A continuación, en la Figura 18, se comprueba mediante la herramienta Burp Suite, que no se logra acceder al archivo robots.txt para verificar el listado de archivos en el directorio principal.

**Figura 18.**

*Información robots.txt – Herramienta Burp Suite*



Fuente: elaboración propia

- **Enumerar aplicaciones en el Servidor Web (OTG-INFO-004)**

**Objetivo de la Prueba**

El objetivo de la prueba es identificar las aplicaciones, que se ejecuta en el servidor web, muchas de las cuales tiene vulnerabilidades y estrategias de ataques conocidas que logran ser explotadas y de esta manera obtener datos importantes de la aplicación web, además, se consigue obtener información del sistema operativo.

**Ejecución de la Prueba**

Para realizar esta prueba, se utiliza el comando nmap, con la siguiente estructura:

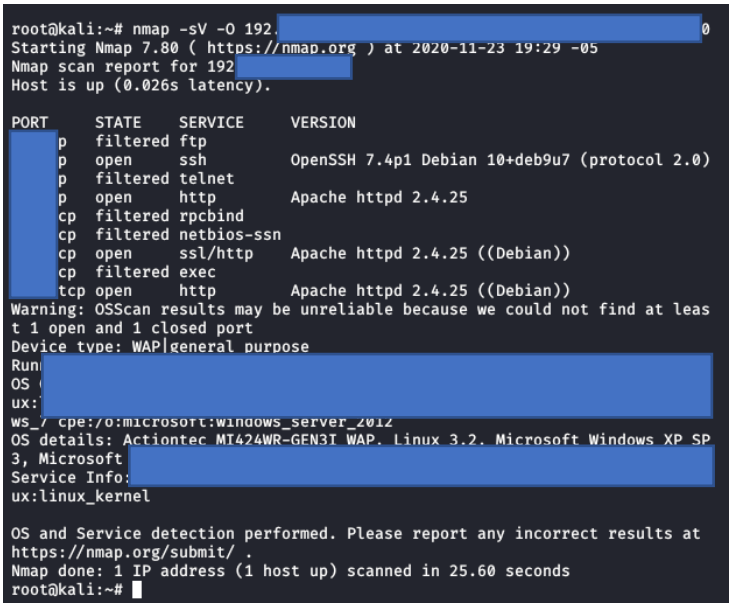
Nmap -sV -O URL/IP -p Puerto/s

- **-sV:** Detectar servicios en ejecución
- **-O:** Versión Sistema Operativo
- **URL:** www.ejemplo.com/192.168...
- **-p:** Puerto/s

A continuación, en la Figura 19, se verifica que los datos obtenidos en relación con los servicios o aplicaciones que se ejecuta en el servidor web.

**Figura 19.**

*Información Servicios o Aplicaciones en ejecución – Herramienta Nmap -sV /Kali Linux*

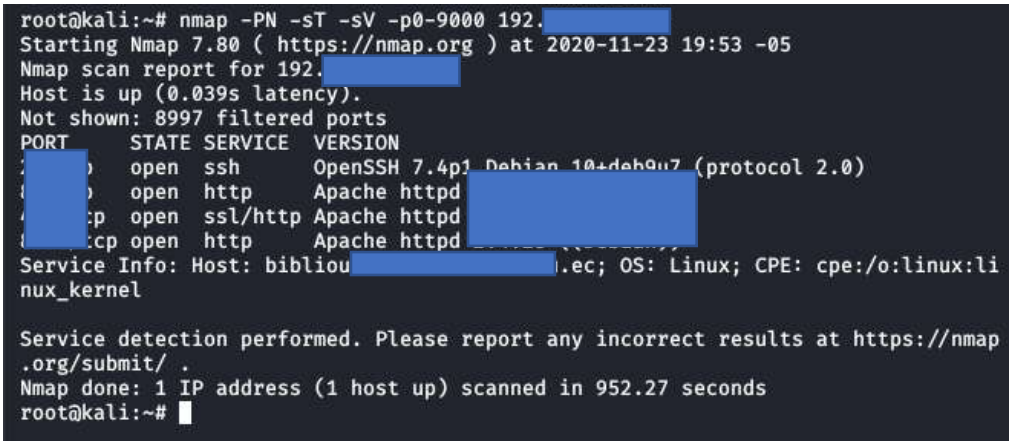



Fuente: elaboración propia

A continuación, en la Figura 20, se denota que los datos obtenidos en relación con los servicios o aplicaciones que se ejecutan en el servidor web utilizan otras opciones que presenta el comando *nmap*.

**Figura 20.**

*Información Servicios o Aplicaciones en ejecución – Herramienta Nmap -PN /Kali Linux*

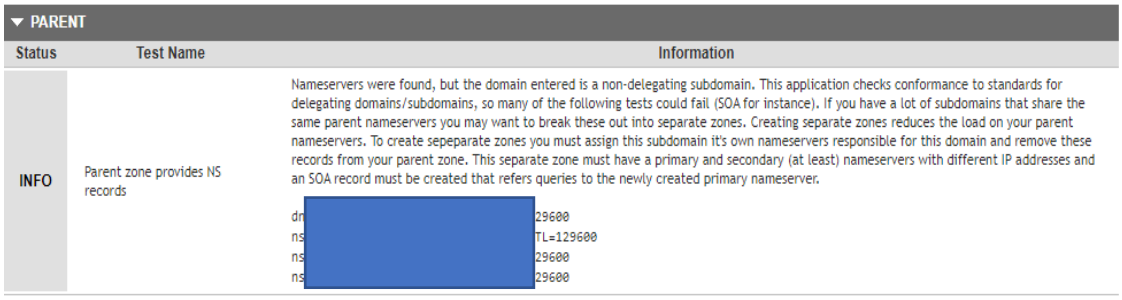



Fuente: elaboración propia

Además, se realiza búsquedas *DNS* basadas en la *web*, como se muestra en la Figura 21, mediante la herramienta online DNSStuff, la cual, contiene datos importantes sobre el DNS de la aplicación web.

**Figura 21.**

*Información DNS Server – Herramienta DNSStuff*

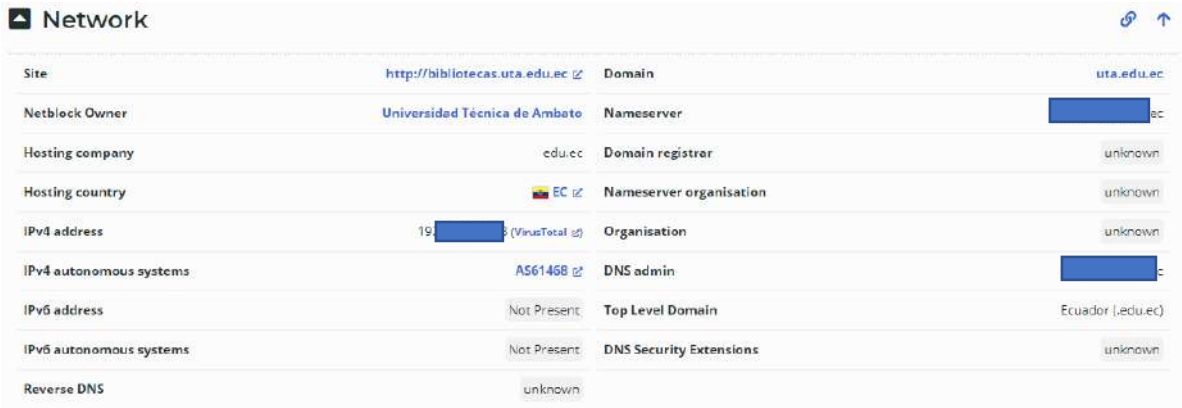



Fuente: elaboración propia

También, se realiza búsquedas *DNS* basadas en la *web*, como se muestra en la Figura 22, mediante la herramienta *online* netcraft de, la cual, se obtiene la información como el dominio, nombre del servidor, registro del dominio, hosting, entre otros.

**Figura 22.**

*Información DNS Server – Herramienta NetCraft*




Fuente: elaboración propia

- **Revisar comentarios en la página web y metadatos por fugas de información (OTG-INFO-005)**

**Objetivo de la Prueba**

El objetivo de la prueba es revisar los comentarios y también los metadatos de la aplicación web, que ayudara a entender mejor la aplicación y así encontrar algún tipo de fuga de información. Se manifiesta que es muy común e incluso recomendable para programadores incluir comentarios detallados y metadatos en el código fuente de una determinada aplicación *web*. Pero por tal situación dichos comentarios y metadatos al ser incluidos en el código HTML estos podrían revelar información importante.

**Ejecución de la Prueba**

Para realizar esta prueba, se va a utilizar la herramienta del navegador View Code.

A continuación, en la Figura 23, se muestra el código fuente de la página.

**Figura 23.**

*Ver Código de Pagina – Herramienta Navegador Opera*

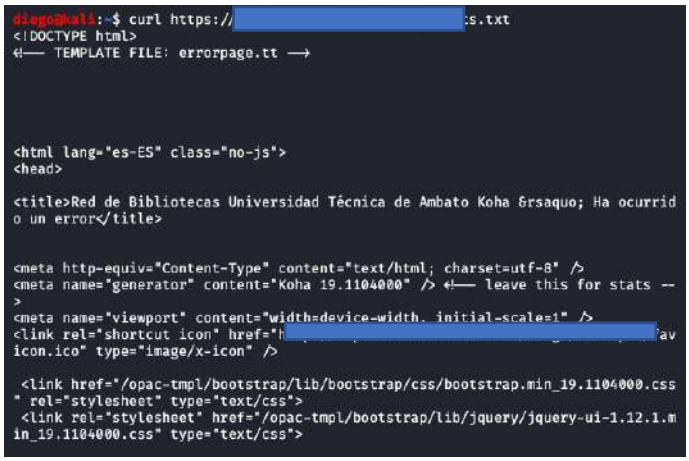
```
<!DOCTYPE html>
<!-- TEMPLATE FILE: opac-main.tt -->



<html lang="es-ES" class="no-js">
<head>

<title>Red de Bibliotecas Universidad Técnica de Ambato Koha</title>


<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta name="generator" content="Koha 19.1104000" /> <!-- leave this for stats -->
<meta name="viewport" content="width=device-width, initial-scale=1" />
<link rel="shortcut icon" href="http://repositorio.uta.edu.ec/image/kohaopac/favicon.ico" type="image/x-icon" />

 <link rel="stylesheet" type="text/css" href="/opac-tmpl/bootstrap/lib/bootstrap/css/bootstrap.min_19.1104000.css">
 <link type="text/css" href="/opac-tmpl/bootstrap/lib/jquery/jquery-ui-1.12.1.min_19.1104000.css" rel="stylesheet">



 <link rel="stylesheet" href="/opac-tmpl/bootstrap/css/opac_19.1104000.css" type="text/css">


<link type="text/css" href="/opac-tmpl/bootstrap/css/print_19.1104000.css" rel="stylesheet" media="print">


 <link rel="unapi-server" type="application/xml" title="unAPI" href="http://19[redacted]3/cgi-bin/koha/unapi" />


<!-- Respond.js brings responsive layout behavior to IE < v.9 -->
<!--[if lt IE 9]>
 <script src="/opac-tmpl/bootstrap/lib/respond.min.js"></script>
<![endif]-->
<script>
    function _(s) { return s } // dummy function for gettext
</script>
<script src="/opac-tmpl/bootstrap/lib/modernizr.min_19.1104000.js"></script>
<link rel="stylesheet" type="text/css" href="/opac-tmpl/bootstrap/lib/font-awesome/css/font-awesome.min_19.1104000.css">
```

Fuente: elaboración propia

También, se realiza la prueba mediante el comando curl, el cual, indica el código fuente de la página web, como se ve en la Figura 24.

**Figura 24.**

*Ver Código de Pagina – Herramienta Curl*



```
diegokali:~$ curl https://[redacted]s.txt
<!DOCTYPE html>
<!-- TEMPLATE FILE: errorpage.tt -->



<html lang="es-ES" class="no-js">
<head>

<title>Red de Bibliotecas Universidad Técnica de Ambato Koha &rsaquo; Ha ocurrid
o un error</title>


<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta name="generator" content="Koha 19.1104000" /> <!-- leave this for stats --
>
<meta name="viewport" content="width=device-width, initial-scale=1" />
<link rel="shortcut icon" href="h[redacted]av
icon.ico" type="image/x-icon" />

 <link href="/opac-tmpl/bootstrap/lib/bootstrap/css/bootstrap.min_19.1104000.css
" rel="stylesheet" type="text/css">
 <link rel="stylesheet" href="/opac-tmpl/bootstrap/lib/jquery/jquery-ui-1.12.1.m
in_19.1104000.css" type="text/css">
```

Fuente: elaboración propia

- **Identificar los puntos de entrada de la aplicación (OTG-INFO-006)**

**Objetivo de la Prueba**

El objetivo de la prueba es entender cómo se forman las solicitudes y las respuestas que se dan mediante la aplicación *web*, que identifican los métodos *GET* y *POST* respectivamente, sus respectivas variables que intervienen en el proceso de intercambio de datos, además, permiten al evaluador identificar probables áreas de debilidad. Ayuda a identificar y mapear las áreas que se encuentran dentro de la aplicación web que se investiga una vez que la enumeración y el mapeo se complete.

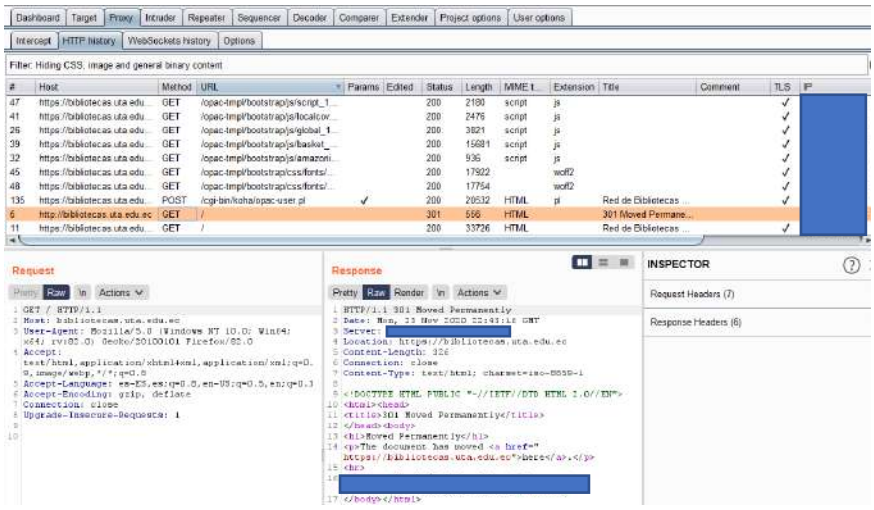
**Ejecución de la Prueba**

Para realizar esta prueba, se utiliza la herramienta Burp Suite, la cual, intercepta la información mediante los métodos *GET* y *POST*.

A continuación, en la Figura 25, se muestra la información interceptada mediante el método *GET,* aplicada a la aplicación web.

**Figura 25.**

*Método GET – Herramienta Burp Suite*

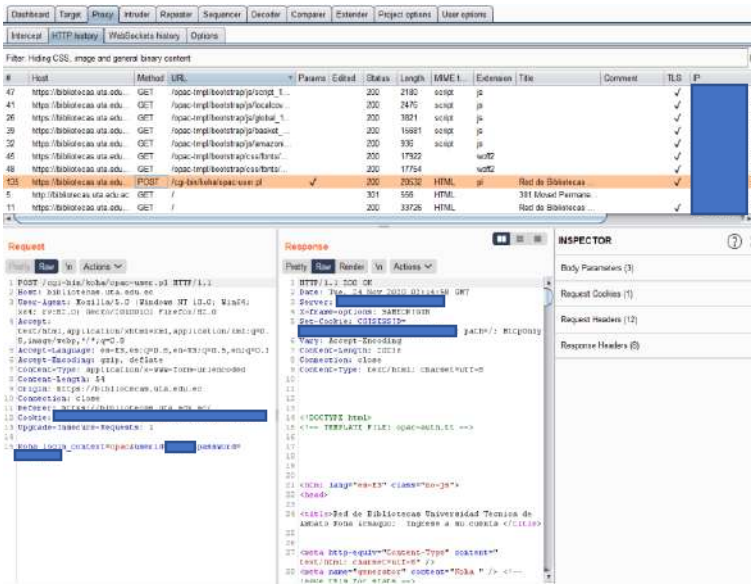



Fuente: elaboración propia

De la misma manera se muestra en la Figura 26, la información interceptada mediante el método *POST*, aplicada a la aplicación web.

**Figura 26.**

*Método POST – Herramienta Burp Suite*



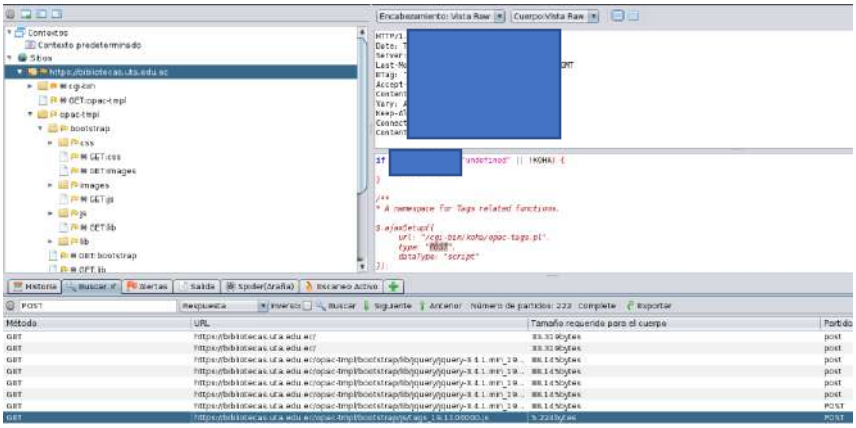
Fuente: elaboración propia

De la misma manera, se realiza una prueba que utiliza la herramienta OWASP ZAP, que al igual que Burp Suite, intercepta la información de una determinada aplicación *web*.

A continuación, en la Figura 27, se muestra el método *GET,* aplicada a la aplicación web.

**Figura 27.**

*Método GET-POST – Herramienta OWASP ZAP*



Fuente: elaboración propia

- **Mapear rutas de ejecución a través de la aplicación (OTG-INFO-007)**

**Objetivo de la Prueba**

El objetivo de la prueba es el crear mapas de la aplicación web de destino y así comprender los principales flujos de trabajo que este realiza.

**Ejecución de la Prueba**

Para realizar esta prueba, se utiliza las herramientas Burp Suite y OWASP Zap, las cuales ayudaran a encontrar archivos y directorios sensibles, que se encuentren dentro la aplicación *web*, cuya finalidad es encontrar si existe algún tipo de fuga de información, este tipo de método se conoce como *web spidering*.

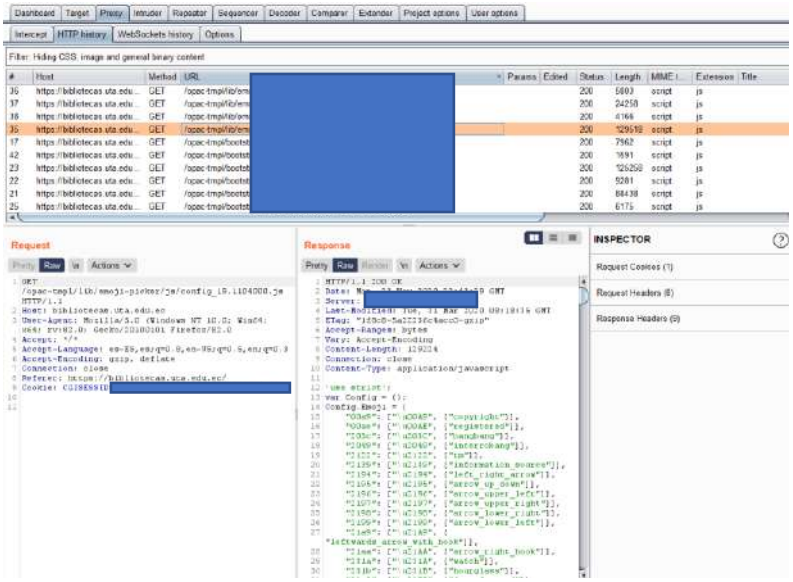
Al realizar esta prueba mediante las herramientas antes mencionadas que interceptan el tráfico de información, se localiza lo siguiente.

- Árbol de Directorios
- Directorios Sensibles
- Archivos Sensibles
- Parámetros y variables en las URL
- Direcciones Administrativas
- Mapa de Sitio
- Interfaces de inicio de sesión
- Metadatos

A continuación, en la Figura 28, se muestra el árbol de directorios en, la cual, se logra desplegar información importante, archivos como *JavaScripts*, son común encontrar en la mayoría de las aplicaciones *web*, estos archivos son una brecha importante para tener en cuenta, con ellos los ciberdelincuentes intentan ejecutar algún código malicioso mediante estos archivos, y así intentar interceptar el sitio o aplicación *web*.

**Figura 28.**

*Árbol de Directorios – Herramienta Burp Suite*

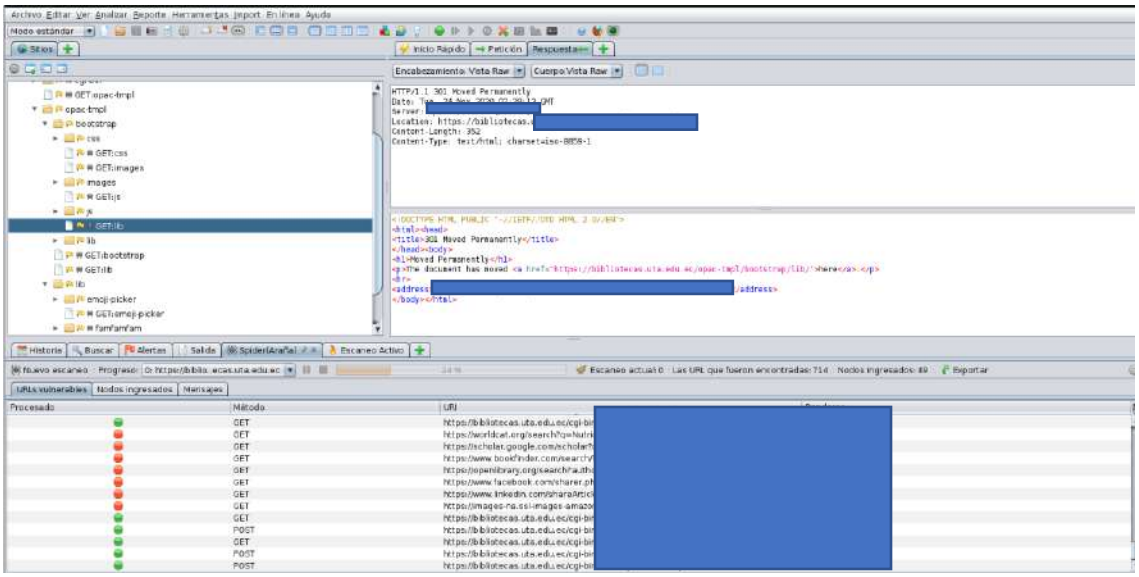


Fuente: elaboración propia

Se tiene también mediante la herramienta OWASP Zap, el árbol de directorios en la Figura 29, la cual, muestra el método web *spidering* que contiene la siguiente información.

**Figura 29.**

*Árbol de Directorios – Método web spidering – Herramienta OWASP Zap*



Fuente: elaboración propia

- **Framework referencial para el uso de huellas digitales en aplicaciones web (OTG-INFO-008)**

**Objetivo de la Prueba**

El objetivo de la prueba es el definir un *framework* que utiliza la aplicación web para la búsqueda de vulnerabilidades, se utilizan varios proveedores y versiones de *frameworks web*, lo que hacen la mayoría de ellos es buscar un marcador desde una ubicación preestablecida y luego compararlo con la base de datos de dichas firmas conocidas.

**Ejecución de la Prueba**

Para realizar esta prueba, se utiliza la herramienta WhatWeb, la cual, provee información importante acerca de las huellas digitales de la aplicación *web*.

A continuación, en la Figura 30, se muestra los datos a través de la página oficial de WhatWeb, la cual, contiene datos principales de la aplicación web, con el servidor, Dirección IP.

**Figura 30.**

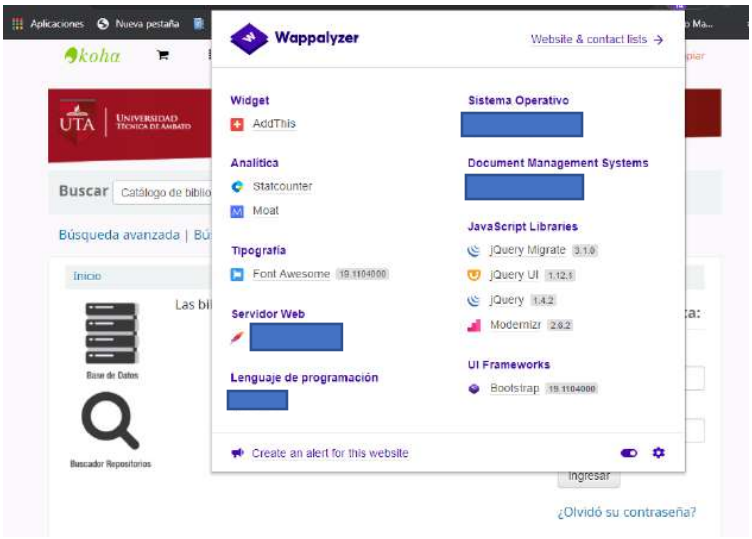
*Fingerprint – Herramienta WhatWeb - online*


Fuente: elaboración propia

En la Figura 31, se observa la siguiente información que utiliza la herramienta *Wappalyzer*, la cual, indica el contenido de las tecnologías que se usan al navegar por la aplicación *web*, la cual, suele resultar de gran beneficio para buscar *frameworks,* que se ejecutan en la aplicación que logran resultar puntos de vulnerabilidad.

**Figura 31.**

*Fingerprint – Herramienta Wappalyzer - online*




Fuente: elaboración propia

- **Aplicación huellas digitales para web (OTG-INFO-009)**

**Objetivo de la Prueba**

El objetivo de la prueba es identificar la aplicación *web* y a su vez la versión, para determinar algunas vulnerabilidades conocidas, la forma apropiada para explotarlas durante la prueba.

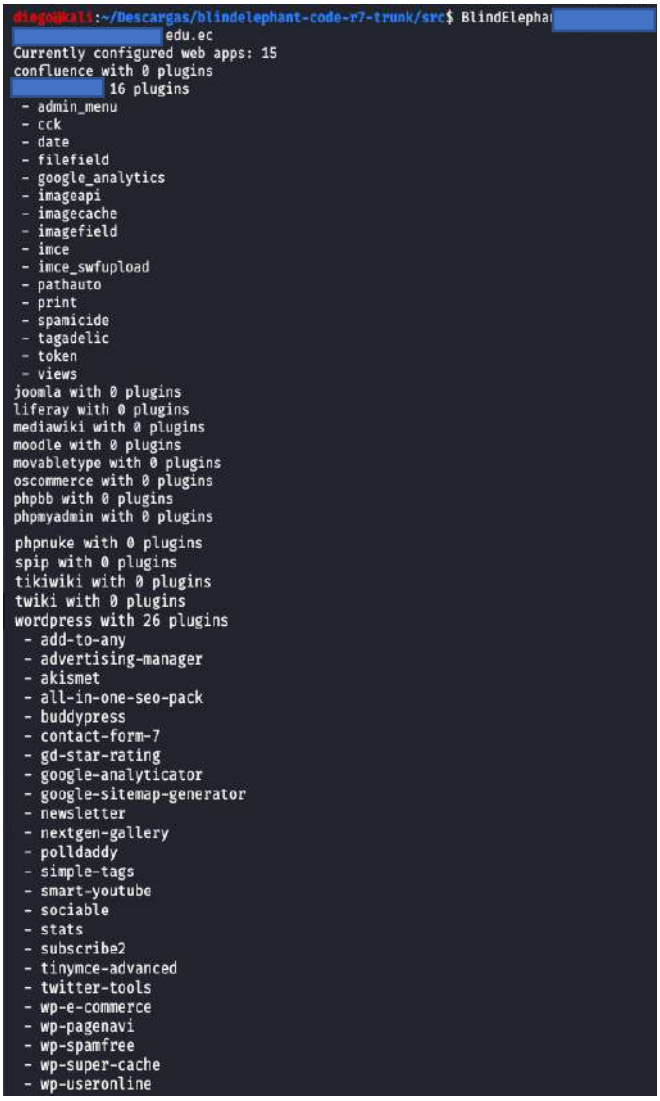
**Ejecución de la Prueba**

Para realizar esta prueba, se utiliza la herramienta BlindElephant, para determinar las vulnerabilidades conocidas.

A continuación, en la Figura 32, se muestra la información sobre los plugin encontrados en los *frameworks Drupal* y Wordpress, asociados a la aplicación *web*.

**Figura 32.**

*Aplicaciones Web– Herramienta BlindElephant*



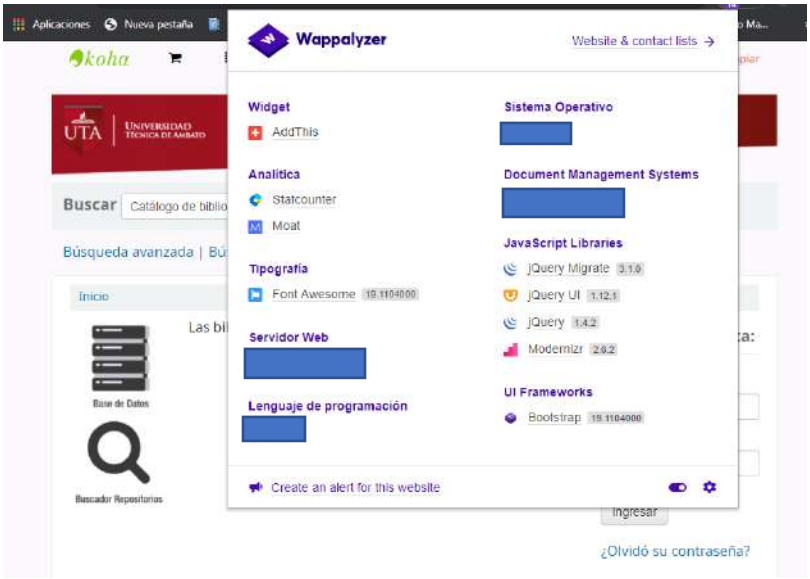

Fuente: elaboración propia

Además, se obtiene importante información mediante la herramienta *Wappalyzer*, la cual, despliega información que obtiene de la aplicación web como, por ejemplo, el *framework* en el que trabaja, lenguaje de programación y sistema operativo del servidor web.

A continuación, se muestra en la Figura 33 las aplicaciones web.

**Figura 33.**

*Aplicaciones Web – Herramienta Wappalyzer - online*




Fuente: elaboración propia

### 2.3.2 Pruebas de gestión de la configuración y la implementación.

Consta de un análisis de la arquitectura que revelan información importante como: código fuente de una aplicación, métodos HTTP permitidos, métodos de autenticación, entre otros.

**OTG-CONFIG-001**

**OTG:** OWASP TEST GUIDE v4.0 (Guía de Pruebas de OWASP v4.0)

**CONFIG:** Se refiere a la categoría de cada fase, en este caso (Pruebas de seguridad a la configuración y despliegue)

**001:** Se refiere al número de categoría a la que pertenece, para el caso es: prueba número uno de la categoría pruebas de seguridad a la configuración y despliegue.

- **Prueba de configuración de la infraestructura de red (OTG-CONFIG-001)**

**Objetivo de la Prueba**

El objetivo de la prueba es conocer la infraestructura de red, conocer como están conectados los diferentes dispositivos, servidores, computadoras, firewall.

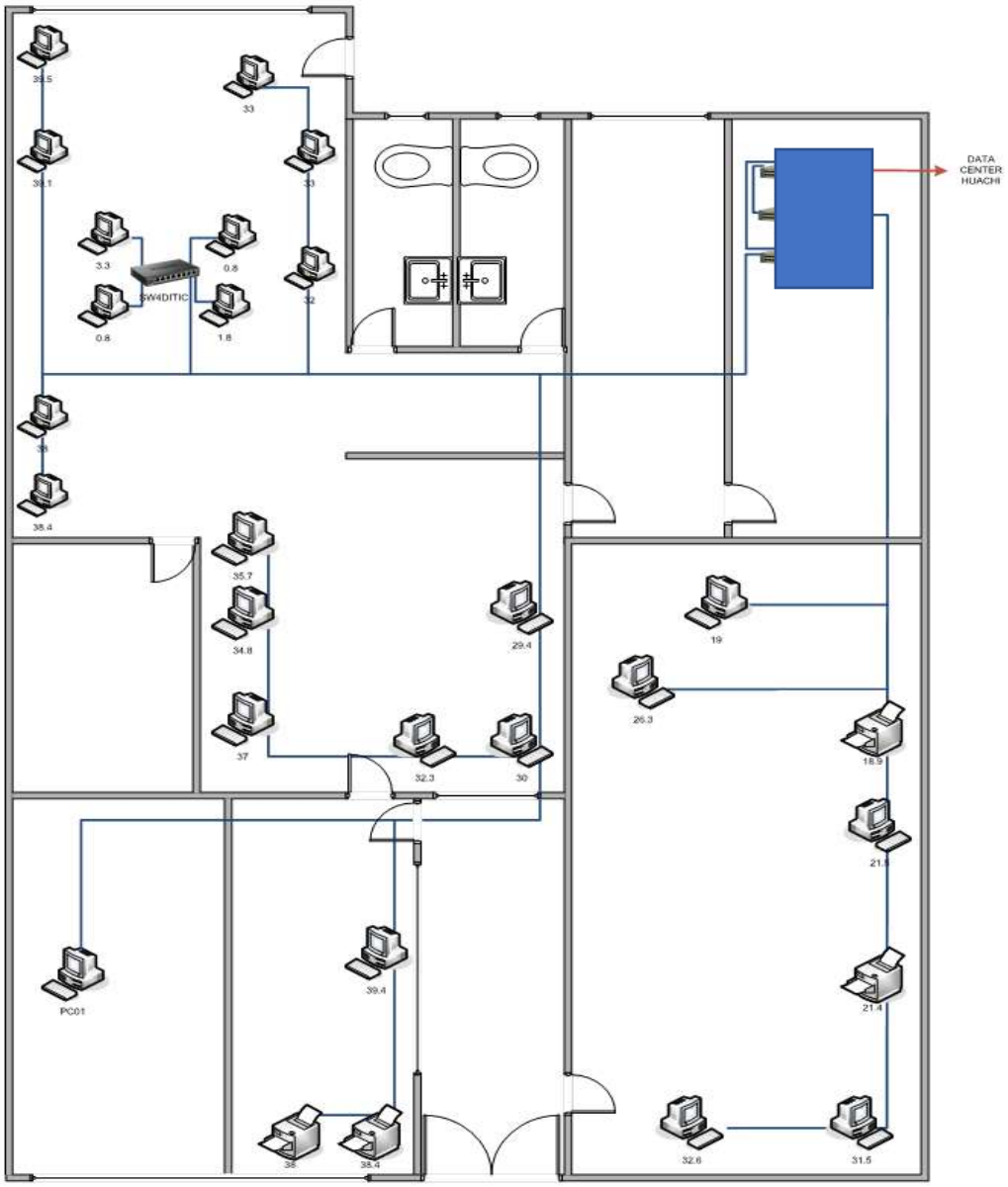
**Ejecución de la Prueba**

Para realizar la comprobación de esta prueba, se obtiene el diagrama de la infraestructura de red e la institución.

A continuación, en la Figura 34, se muestra como están conectados los diferentes dispositivos a la red de la institución.

**Figura 34.**

*Diagrama de infraestructura de red de la Institución*




Fuente: institución

- **Configuración de la plataforma de aplicaciones de prueba (OTG-CONFIG-002)**

**Objetivo de la Prueba**

El objetivo de la prueba es conocer si las instalaciones que se han realizado dentro de la aplicación fueron de manera manual o si se han realizado instalaciones automáticas.

**Ejecución de la Prueba**

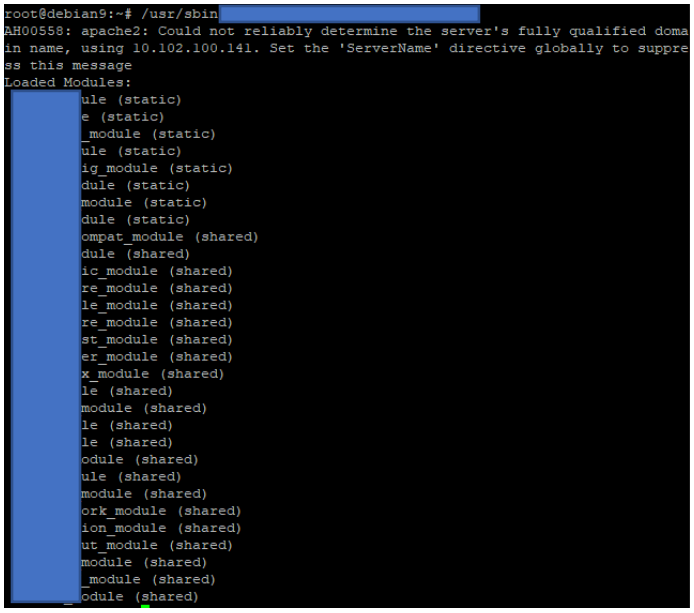
Para realizar la comprobación de esta prueba, se utilizan los comandos, los cuales listan los módulos activos en el servidor *web*, para esta prueba no es necesaria la IP de la aplicación *web*.

- apache2 -L
- apachectl -M
- apche2ctl -t -D DUMP_MODULES
- httpd -M
- /usr/local/apache22/bin/httpd -M

A continuación, la Figura 35 muestra los módulos activos dentro del servidor *Apache*.

**Figura 35.**

*Módulos Activos Servidor Web – Herramienta – Apache2ctl*




Fuente: elaboración propia

- **Manejo de extensiones de archivo de prueba para información confidencial (OTG-CONFIG-003)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar como los servidores manejan las peticiones a las diferentes extensiones, ayuda a comprender mejor el comportamiento del servidor.

**Ejecución de la Prueba**

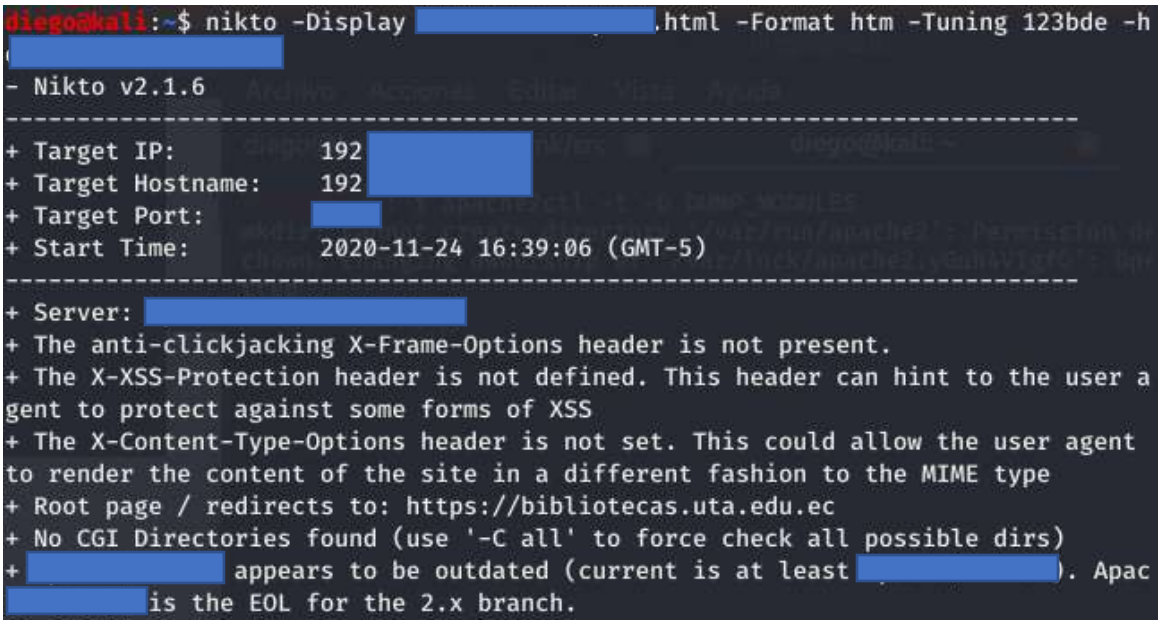
Para realizar la comprobación de esta prueba, se utiliza la herramienta Nikto mediante el comando.

**nikto -Display 1234EP -o report.html -Format htm -Tuning 123bde -host IP**

A continuación, en la Figura 36, se muestra los principales datos como la Dirección ip, Hostname y el puerto en el que se ejecuta la aplicación, que se obtiene con la herramienta Nikto.

**Figura 36.**

*Servidores – Manejo de Peticiones – Herramienta Nikto*

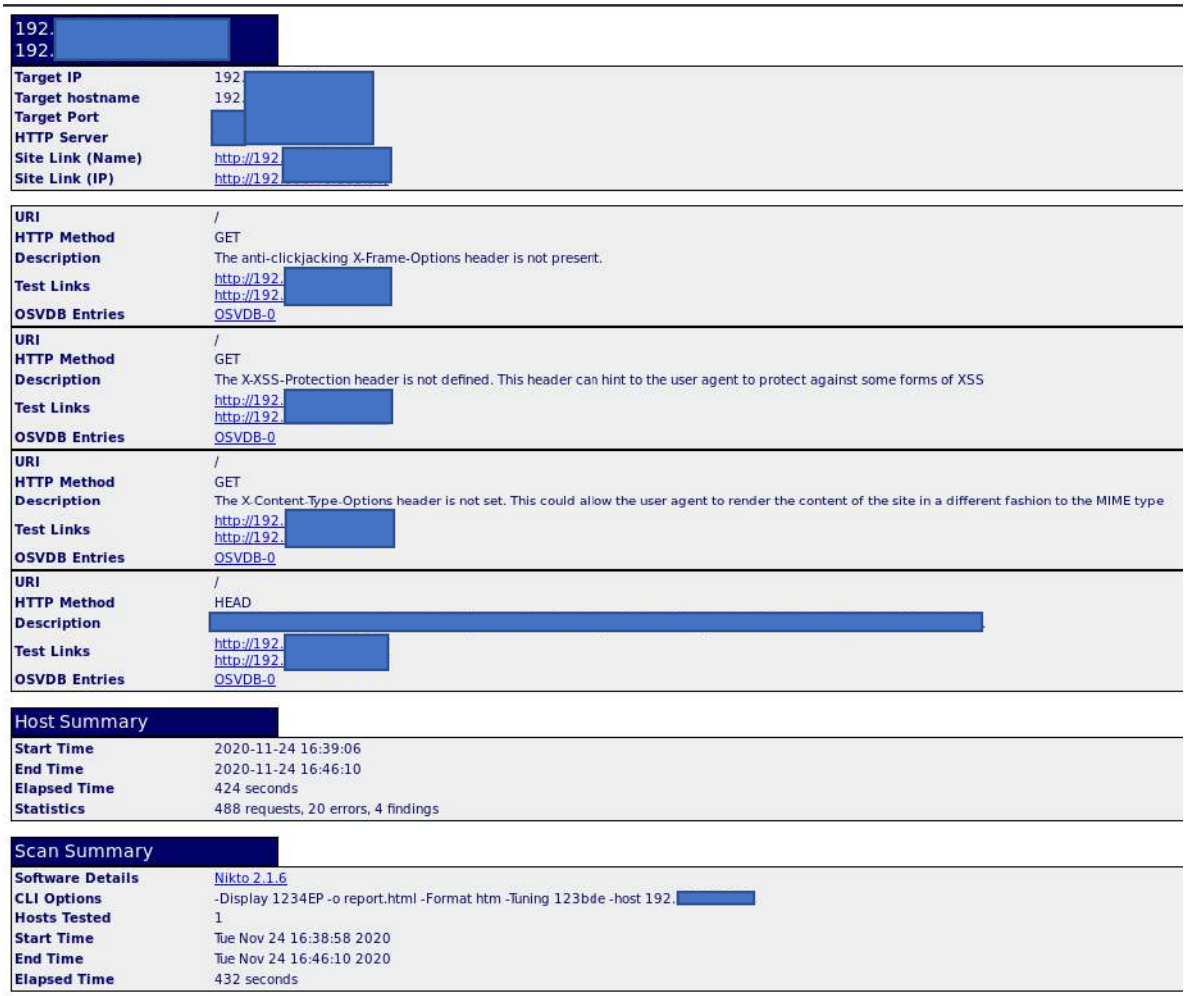



Fuente: elaboración propia

En la Figura 37, se muestra el reporte en formato html obtenido mediante la herramienta Nikto.

**Figura 37.**

*Servidores – Reporte de Manejo de Peticiones – Herramienta Nikto*



| 192. [redacted] | |
|---|---|
| 192. | |
| **Target IP** | 192. |
| **Target hostname** | 192. |
| **Target Port** | |
| **HTTP Server** | |
| **Site Link (Name)** | http://192 |
| **Site Link (IP)** | http://192 |

| | |
|---|---|
| **URI** | / |
| **HTTP Method** | GET |
| **Description** | The anti-clickjacking X-Frame-Options header is not present. |
| **Test Links** | http://192.<br>http://192. |
| **OSVDB Entries** | OSVDB-0 |

| | |
|---|---|
| **URI** | / |
| **HTTP Method** | GET |
| **Description** | The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS |
| **Test Links** | http://192.<br>http://192. |
| **OSVDB Entries** | OSVDB-0 |

| | |
|---|---|
| **URI** | / |
| **HTTP Method** | GET |
| **Description** | The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type |
| **Test Links** | http://192.<br>http://192. |
| **OSVDB Entries** | OSVDB-0 |

| | |
|---|---|
| **URI** | / |
| **HTTP Method** | HEAD |
| **Description** | |
| **Test Links** | http://192.<br>http://192. |
| **OSVDB Entries** | OSVDB-0 |

| Host Summary | |
|---|---|
| **Start Time** | 2020-11-24 16:39:06 |
| **End Time** | 2020-11-24 16:46:10 |
| **Elapsed Time** | 424 seconds |
| **Statistics** | 488 requests, 20 errors, 4 findings |

| Scan Summary | |
|---|---|
| **Software Details** | Nikto 2.1.6 |
| **CLI Options** | -Display 1234EP -o report.html -Format htm -Tuning 123bde -host 192. |
| **Hosts Tested** | 1 |
| **Start Time** | Tue Nov 24 16:38:58 2020 |
| **End Time** | Tue Nov 24 16:46:10 2020 |
| **Elapsed Time** | 432 seconds |

Fuente: elaboración propia

- **Revisar archivos antiguos, de copia de seguridad y sin referencia en busca de información confidencial (OTG-CONFIG-004)**

### Objetivo de la Prueba

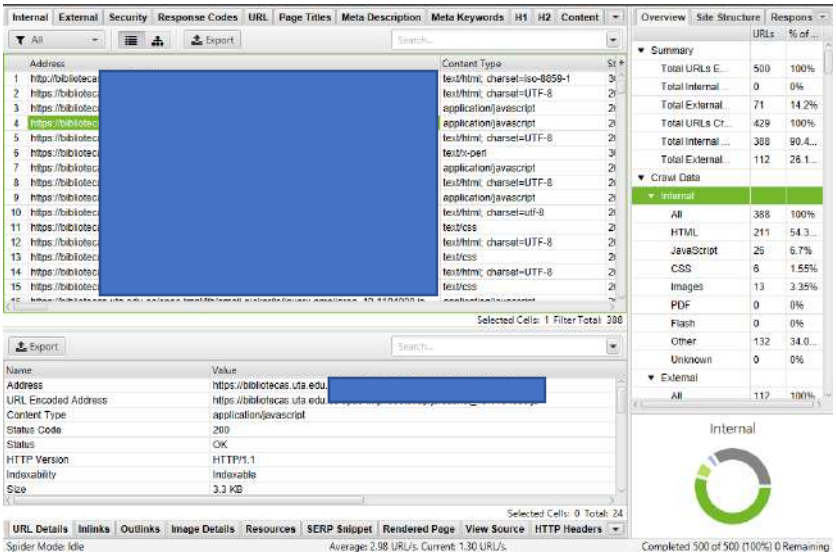
El objetivo de la prueba es verificar archivos viejos que no se encuentren referenciados, los cuales contienen datos sensibles, además, existen archivos que se crean como consecuencia de editar otros archivos, los cuales generalmente contienen la misma información que el archivo original.

**Ejecución de la Prueba**

Para realizar la comprobación de esta prueba, se utiliza la herramienta Screaming Frog SEO Spider, para obtener toda la información de los directorios que se encuentran dentro de la aplicación *web*. A continuación, en la Figura 38, se muestra la información de los directorios que se encuentran dentro de la aplicación web.

**Figura 38.**

*Árbol de Directorios – Herramienta Screaming Frog SEO Spider*




Fuente: elaboración propia

La finalidad de esta prueba es recopilar información y determinar si existen archivos que alcancen a revelar información confidencial, la cual, logra afectar el desempeño de la aplicación y de otras que, además, se encuentran de alguna manera relacionadas institucionalmente o, así también, ser perjudicial para los usuarios de esta.

- **Infraestructura de enumeración e interfaces de Administración de Aplicaciones (OTG-CONFIG-005)**

**Objetivo de la Prueba**

El objetivo de la prueba es encontrar interfaces de administrador, las cuales están presentes directamente en la aplicación o en el servidor donde se aloja la aplicación, mediante estas interfaces

permiten al usuario que tienen privilegios hacer realizar actividades que los usuarios no autorizados no los consiguen realizar.

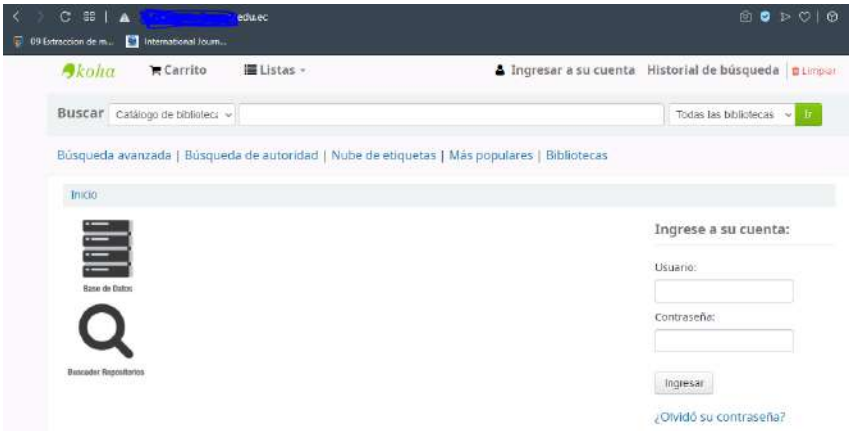
**Como comprobar**

Para realizar la comprobación de esta prueba, se utilizará directamente la URL de la aplicación *web*, la cual, tiene como *framework* principal *Koha*, la misma que logra acceder a su parte administrativa de la siguiente manera.

A continuación, en la Figura 39, se muestra cómo se consigue acceder a la parte administrativa de la aplicación *web,* ingresa un usuario y una contraseña.

**Figura 39.**

*Administración de la Aplicación Web*




Fuente: elaboración propia

- **Prueba de Métodos HTTP (OTG-CONFIG-006)**

**Objetivo de la Prueba**

El objetivo de la prueba identificar los métodos que la aplicación web maneja en el servidor, HTTP *(Hypertext Transfer Protocol)* maneja métodos de *GET* y *POST*, estos serán configurados de manera correcta, pues suelen ser estos métodos utilizados para fines delictivos por parte de los ciberdelincuentes, algunos de estos métodos se logra plantear un potencial riesgo para la aplicación *web*, y para las demás aplicaciones que se manejan dentro de la Institución.

**Ejecución de la Prueba**

Para realizar la comprobación de esta prueba, se utiliza la herramienta nmap, la cual, ayuda a obtener los métodos manejados por el protocolo HTTP, hacia los puertos 80 y 443, mediante el siguiente comando.

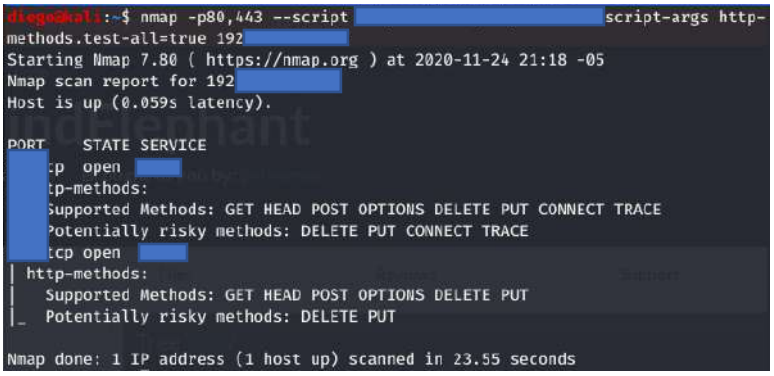
**nmap -p80,443 --script http-methods, http-trace --script-args http-methods.test-all=true IP**

- **-p80:** Puertos a buscar
- **--script http-methods:** Script nmap para identificar metodos HTTP
- **http-trace:** Muestra si el método TRACE está habilitado, si la depuración está habilitada, devuelve los campos de encabezado que se modificaron en la respuesta.
- **--script-args http-methods.test:** Proporciona argumentos a los scripts.

A continuación, se muestra en la Figura 40, los métodos que se ejecutan en el servidor web de la aplicación.

**Figura 40.**

*Métodos HTTP – Herramienta Nmap*




Fuente: elaboración propia

- **Prueba de Seguridad de Transporte Estricto HTTP - HSTS (OTG-CONFIG-007)**

**Objetivo de la Prueba**

El objetivo de la prueba es probar si existe la presencia del encabezado HSTS *(Strict Transport Security)* que es un mecanismo de política de seguridad web, el cual, ayuda a proteger los sitios o

aplicaciones web contra ataques informáticos, por ejemplo, ataques de degradación de protocolo y secuestro de cookies, lo cual, se realiza con la comprobación de la existencia de la Rubrica HSTS en la respuesta del servidor de un proxy, o a su vez, se utiliza el comando *curl*.

**Ejecución de la Prueba**

Para realizar la comprobación de esta prueba, se utiliza el comando curl para determinar el encabezado HSTS, el cual, es el siguiente:

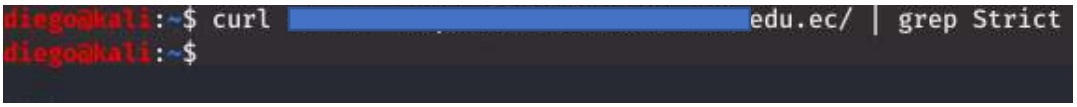
**curl -s -D- https://bibliotecas.uta.edu.ec/ | grep Strict**

- **-s:** Parámetro de modo silencioso. No muestre el medidor de progreso ni los mensajes de error. Silencia a Curl.
- **-D-:** Escriba los encabezados de protocolo recibidos en el archivo especificado.
- **| grep Strict:** Busca la palaba Stric dentro del encabezado.

A continuación, en la Figura 41, se muestra mediante el comando antes mencionado, se aplica a la aplicación web, pero no devuelve ninguna información, esto quiere decir que no se aplica este encabezado de seguridad a dicha aplicación.

**Figura 41.**

*Encabezado HSTS – Herramienta Curl*




Fuente: elaboración propia

En la Figura 42 también, se verifica el encabezado HSTS para la misma aplicación *web*, y se logra confirmar de la misma manera, que no se aplica este mecanismo importante de seguridad para aplicaciones *web*.

**Figura 42.**

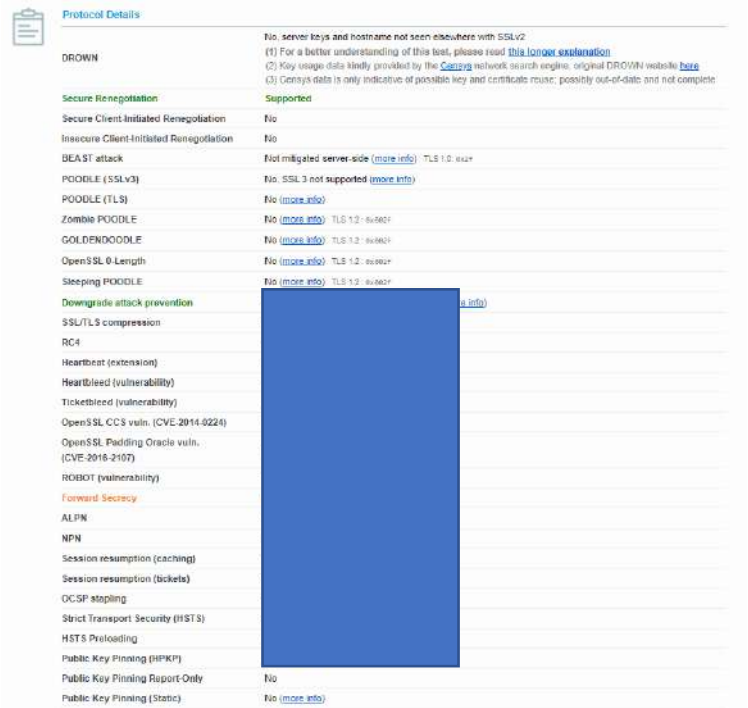
*Encabezado HSTS – Herramienta hstspreload - online*


Fuente: elaboración propia

Además, se realizó otro tipo de prueba mediante la aplicación Quaks.org, que es un *tester* de HSTS online, el cual, arrojo el siguiente resultado como se muestra en la Figura 43, en el cual, se determina que no se ejecuta HSTS.

**Figura 43.**

*Encabezado HSTS – Herramienta Qualys SSL Labs - online*




Fuente: elaboración propia

- **Prueba de Política de Dominio cruzado RIA (OTG-CONFIG-008)**

Ria *(Rich Internet Applications),* son aplicaciones basadas en la Web que tienen algunas características de las aplicaciones gráficas de escritorio, adopto las características de políticas de crossdomain.xml, esto quiere decir que un dominio logra conceder acceso remoto a los servidores desde otro dominio totalmente diferente.

**Objetivo de la Prueba**

El objetivo de la prueba es buscar la configuración del archivo crossdomain.xml, el cual, si se encuentra mal configurado este logra generar ataques *Cross Site Request Forgery*, y permitir así a que personas no autorizadas a servicios y datos importantes dentro del servidor.

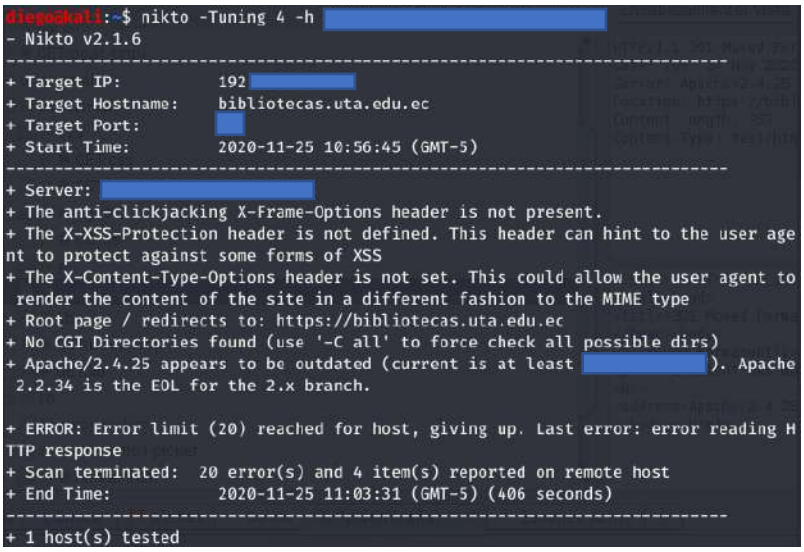
**Ejecución de la Prueba**

Para realizar la comprobación de esta prueba, se utiliza la herramienta nikto, la cual, permite identificar si existe o no la configuración XSS *(Cross-Site Scripting)* presente en el servidor.

A continuación, en la Figura 44, se muestra el resultado del escaneo realizado a la aplicación y determina si existe o no una configuración para *Cross-Site*.

**Figura 44.**

*XSS Cross-Site – Herramienta Nikto*




Fuente: elaboración propia

### 2.3.3 Pruebas de Gestión de Identidad

En esta sección de pruebas, se verifica los diferentes roles de usuario y se realiza pruebas del proceso de registro. Recuperación de los detalles de la cuenta de usuario por registro o inicios de sesión fallidos, también, se prueba la política de registro de nombre de usuario.

**OTG-IDENT-001**

**OTG:** OWASP TEST GUIDE v4.0 (Guía de Pruebas de OWASP v4.0)
**IDENT:** Se refiere a la categoría de cada fase, en este caso (Pruebas de Gestión de Identidad)
**001:** Se refiere al número de categoría a la que pertenece, para el caso es: prueba número uno de la categoría pruebas de Gestión de Identidad.

- **Prueba de definición de Roles (OTG-IDENT-001)**

**Objetivo de la Prueba**

El objetivo de la prueba es conocer los diferentes procesos para la creación de roles, las cuales para el caso de la aplicación *web*, no se crean roles a usuario externos, puesto que, ya se ha creado previamente los mismos.

Se concluye que, para las aplicaciones o sitios web, la autorización se vuelve obligatoria, puesto que, no es la única manera de gestionar el acceso mediante permisos a los recursos de un determinado sistema. En entornos más confiables, en la que confidencialidad de la información no es crítica, controles menos robustos como, por ejemplo, el flujo de trabajo de aplicación y el registro de auditoría alcanzan a cubrir requisitos de integridad de los datos, mientras no se restrinja el acceso al usuario, consiguen convertirse en un paso importante para la inseguridad de una determinada aplicación web.

**Ejecución de la Prueba**

A continuación, en la Tabla 6, se identifican los roles con los que cuenta la aplicación *web*.

**Tabla 6.**

*Roles de Aplicación Web de Bibliotecas*

| Rol | Activo |
|---|---|
| *Administrator* | Si |
| *Student* | Si |
| *Teacher* | Si |
| *Juvenile* | No |
| *Staff* | No |

Fuente: elaboración propia

- **Prueba de Registro de Usuarios (OTG-IDENT-002)**

**Objetivo de la Prueba**

El objetivo de la prueba es conocer los diferentes procesos de creación de usuarios, ciertas aplicaciones *web*, realizan este tipo de procesos, de manera automática o semi-automatica, muchas aplicaciones realizan este proceso de forma automática. Manejar datos de usuarios de forma manual, se convierte en un proceso bastante complejo de administrar por lo que hoy en día, no se realiza.

- La verificación de los requisitos de identidad para registro de usuarios esté o presente y alineados con los requerimientos de seguridad y negocio.
- Validar el proceso de registro.

**Ejecución de la Prueba**

Para realizar esta prueba, se responderá las siguientes preguntas y se comprueba cada una de ellas.

La verificación de los requisitos de identidad para registro de usuarios esté o presente y alineados con los requerimientos de seguridad y negocio.

- **¿Cualquier persona consigue registrarse para acceder?**

  Para el caso de la aplicación web, no se cuenta con una opción para realizar el registro de un usuario, por lo que los estudiantes logren acceder únicamente con las credenciales otorgadas por la Institución.

- **¿Son validados por un ser humano antes de crear los registros, o, se conceden automáticamente si se cumplen los criterios?**

  Para el inicio de sesión, se cuenta con un control de usuario no autorizados a la aplicación *web*.

- **¿Podría la misma persona o identidad registrarse varias veces?**

  Para este caso el usuario si consigue registrarse varias veces, la institución maneja credenciales que son únicas y por estudiante.

- **¿Podrían registrarse usuarios para diferentes roles o permisos?**

  La aplicación web cuenta con varios roles para que un usuario consiga acceder.

- **¿Qué documento de identidad se requiere para que un registro tenga éxito?**

  Se utiliza, la Cédula y el Pin del estudiante.

- **¿Son las identidades registradas verificadas?**

  Se verifica previamente los datos de usuario, pero no cuenta con un control de quien accede a la aplicación *web*, es decir, validado por un *captcha.*

A continuación, se muestra en la Figura 45, el inicio de sesión de la aplicación *web*.

**Figura 45.**

*Inicio de sesión de la Aplicación Web*


Fuente: elaboración propia

A continuación, se muestra en la Figura 46, el inicio de sesión con un usuario que no cuenta con las credenciales para acceder a la aplicación *web*.

**Figura 46.**

*Inicio de sesión de la Aplicación Web – sin credenciales*


Fuente: elaboración propia

Validar el proceso de registro

- ✓ **¿Logra la información de identidad ser fácilmente falsificada?**
  La información ingresada para para ser autenticados en la aplicación, no es fácil de falsificar, puesto que al momento de ingresar datos erróneos la aplicación emite una alerta que indica que los datos no son correctos, y es un usuario común, no podría vulnerar el acceso a esta.
- ✓ **¿Logra el intercambio de información durante el registro ser manipulado?**
  No se logra intercambiar información durante el registro puesto que la aplicación no cuenta con un método de registro de usuarios externos a la institución.

- **Prueba de Creación de Cuentas (OTG-IDENT-003)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar que las cuentas asociadas a la aplicación *web* son correctas y cuenten la seguridad respectiva para la creación de estas.

**Ejecución de la Prueba**

Para realizar esta prueba, se cuenta con un módulo para la creación de una determinada cuenta, el módulo no tiene este apartado, por lo tanto, esta prueba se ha visto obviada.

- **Prueba de enumeración de cuentas y cuentas de usuario adivinables (OTG-IDENT-004)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar las aplicaciones web, en muchos casos revelan cuando existe un nombre de usuario en el sistema, esto se da como consecuencia de la mala configuración o como una decisión de diseño, al momento de desarrollar un sistema web.

**Ejecución de la Prueba**

Para realizar esta prueba, se cuenta con un módulo para la creación de una determinada cuenta, el módulo no cuenta con este apartado, por lo tanto, esta prueba se ha visto obviada.

### 2.3.4 Pruebas de Autenticación

En esta sección de pruebas, se alcanza a decir que Autenticación es establecer o confirmar algo como auténtico, es decir, que las afirmaciones hechas sobre alguna cosa son verdaderas. Autenticar un objeto significa confirmar su procedencia, mientras que si se habla de autenticar a un usuario o persona generalmente consiste en verificar su identidad. La autenticación depende de uno o más factores para su validación.

En seguridad informática, la autenticación es el proceso de tratar de verificar la identidad digital del remitente de una comunicación. Un ejemplo de este determinado proceso es el de inicio de sesión. Probar un esquema de autenticación figura comprender cómo funciona el proceso de autenticación y usar esta información para evitar el mecanismo de autenticación.

**OTG-AUTHN-001**

**OTG:** OWASP TEST GUIDE v4.0 (Guía de Pruebas de OWASP v4.0)
**AUTHN:** Se refiere a la categoría de cada fase, en este caso (Pruebas de Autorización)
**001:** Se refiere al número de categoría a la que pertenece, para el caso es: prueba número uno de la categoría pruebas de Autorización.

- **Prueba del Trasporte de Credenciales en un canal encriptado (OTG-AUTHN-001)**

**Objetivo de la Prueba**

El objetivo de la prueba, se la realiza con la finalidad de asegurarse que un ciberdelincuente no consiga obtener información sensible simplemente, se rastrea en la red con una herramienta de ataque de vulnerabilidades, la cual, logra ser un *sniffer*.

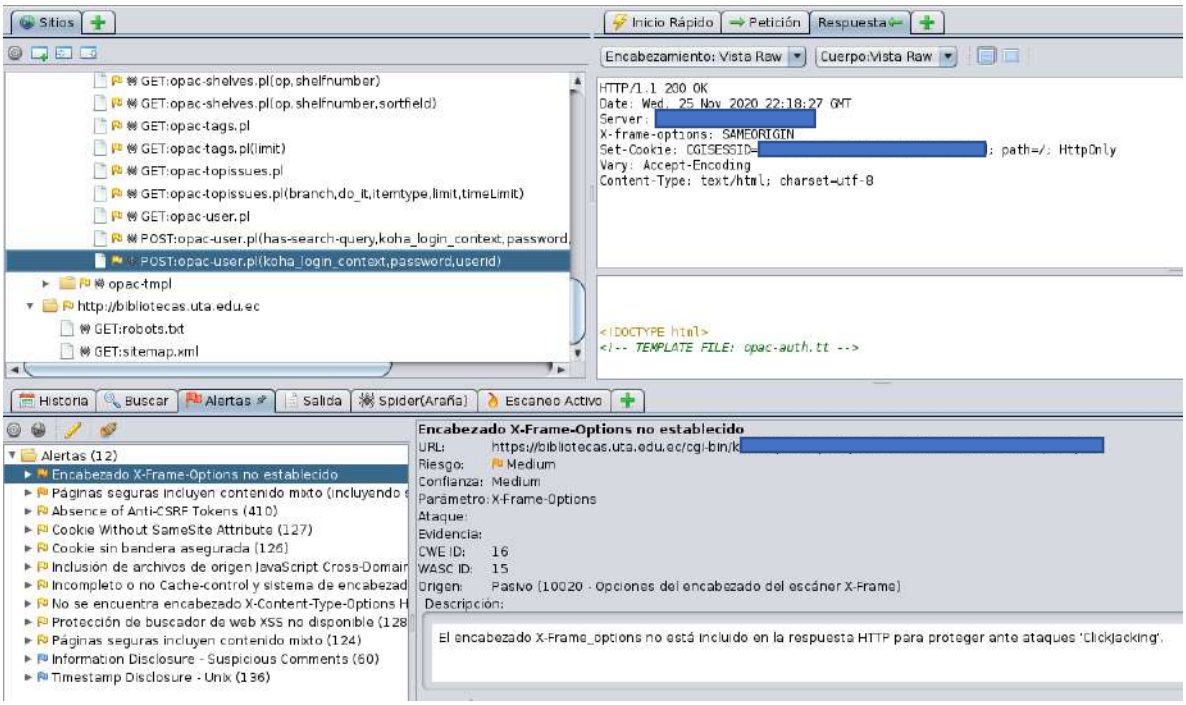
**Ejecución de la Prueba**

Para realizar esta prueba, se usa la herramienta WebScarab para capturar los encabezados de los paquetes e inspeccionarlos.

A continuación, en la Figura 47, se muestra el resultado de la inspección que se realiza mediante la herramienta OWASP Zap para el análisis de los encabezados en, la cual, se verifica que existe un encabezado relacionado con la autenticación.

**Figura 47.**

*Encabezados de Paquetes – Herramienta OWASP Zap*

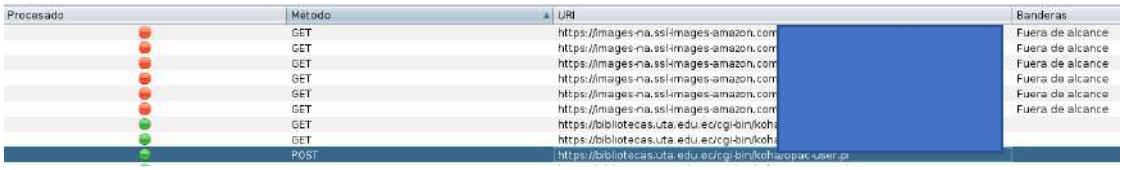



Fuente: elaboración propia

De la misma manera en la Figura 48, se muestra el valor que se envía mediante el método *POST* el formulario de autenticación.

**Figura 48.**

*Encabezados de Paquetes – Método POST – Herramienta OWASP Zap*




Fuente: elaboración propia

- **Prueba de Credenciales por defecto (OTG- AUTHN-002)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar que las credenciales hoy en día no se configuran correctamente y las credenciales predeterminadas proporcionadas para la autenticación inicial y configuración nunca son cambiadas, es por ello, que se verificara esta prueba. Credenciales predeterminadas son bien conocidas por los evaluadores de penetración y, es decir, por atacantes maliciosos que alcanzan a utilizarlas para obtener acceso a varios tipos de aplicaciones web.

**Ejecución de la Prueba**

Para realizar esta prueba, se utiliza una herramienta para realizar fuerza bruta, por lo que no se realiza la comprobación en la aplicación web, con la finalidad de no exponer usuarios y credenciales.

- **Prueba para determinar un mecanismo de bloqueo débil (OTG-AUTHN-003)**

**Objetivo de la Prueba**

El objetivo de la prueba es cubrir todos los aspectos de autenticación donde los mecanismos de bloqueo serían apropiados, por ejemplo, cuando al usuario se le presenten preguntas de seguridad al olvidar su contraseña, al momento de autenticarse.

Mediante un mecanismo de bloqueo fuerte, la aplicación no sería susceptible a ataques de fuerza bruta. Posterior a un ataque de fuerza bruta, un usuario malintencionado podría tener acceso a información importante de la aplicación web, e incluso de la Institución.

**Ejecución de la Prueba**

Para realizar esta prueba, se evalúan las siguientes características:

• Evaluar la capacidad del mecanismo de bloqueo de cuentas para mitigar el ingreso forzado adivinanza de contraseñas.

• Evaluar la resistencia del mecanismo de liberación para abrir sin autorización la cuenta.

Para determinar lo que se va a evaluar se realiza las siguientes preguntas:

- **¿Cuál es el riesgo de forzado o adivinanza de contraseñas en la aplicación?**
  Seria romper la principal seguridad al primer acceso a una aplicación *web*, de lo cual, un atacante consigue obtener información sensible, y posterior a ello romper de manera fácil las próximas seguridades que presente una aplicación *web*.
- **¿Basta un CAPTCHA para mitigar este riesgo?**
  Existen varios métodos de autenticación para aplicaciones *web*, no es suficiente aplicar un *captcha,* pero si es importante, ayuda a proteger de alguna manera el acceso a la aplicación web, además, se podría incluir preguntas de seguridad, o validaciones adicionales que ayuden a proteger aún más el acceso.
- **El número de intentos de registro fallidos antes del bloqueo. Si el umbral de bloqueo es bajo, entonces los usuarios válidos logran ser bloqueados a menudo. Si el umbral de bloqueo es alto, entonces el atacante tiene más intentos para forzar la cuenta antes de que se bloquee. ¿Depende del propósito de la aplicación, un rango entre cinco a diez intentos sin éxito es un umbral de bloqueo típico?**
  Podría ser una lógica de negocio de quien va a utilizar la aplicación, o, a qué tipo de usuarios se va a destinar el manejo de la aplicación, por ello, se alcanza a estimar que será un rango valido entre tres y cinco ocasiones.
- **¿Cómo se desbloquean las cuentas?**
  Las cuentas se desbloquean mediante el envío de correo electrónico en, la cual, se detalla el motivo por, el cual, se desbloqueará la cuenta y posterior a ello el administrador del sistema desbloquea dicha cuenta solicitada.

- **Prueba para eludir el esquema de autenticación (OTG-AUTHN-004)**

**Objetivo de la Prueba**

El objetivo de la prueba es tratar de que se consiga modificando el parámetro de *URL* determinado, mediante la manipulación de la forma o por falsificación de las sesiones, se rompa la autenticación, obviando el registro en la página principal y llamado directamente a una página externa que se supone accedería únicamente después que se realiza la autenticación correctamente.

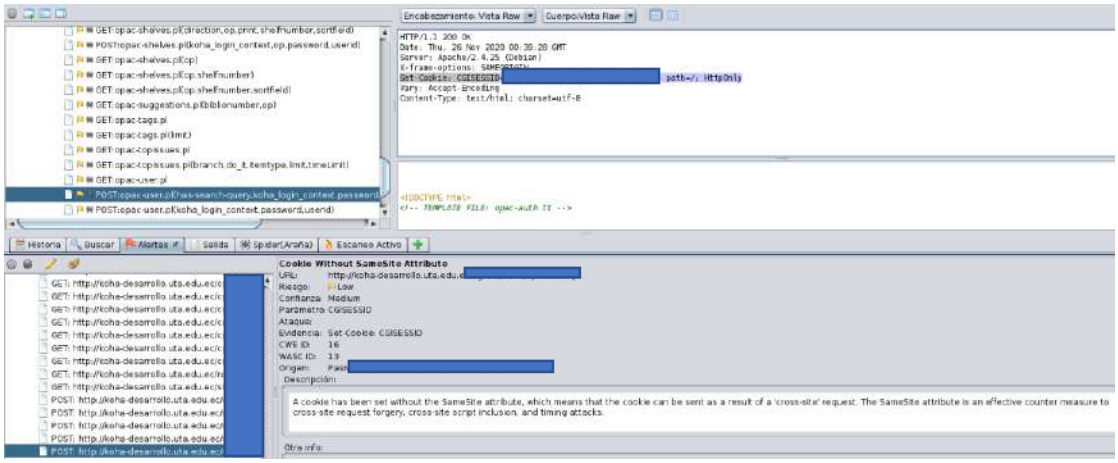
**Como comprobar**

Esta prueba se la realiza mediante la herramienta de OWASP ZAP, para interceptar el acceso o autenticación a la aplicación *web*.

A continuación, en la Figura 49, se muestra la intercepción del acceso a la aplicación mediante el *ID* de sesión de un usuario cualquiera.

**Figura 49.**

*Intercepción de Autenticación – Herramienta OWASP Zap*

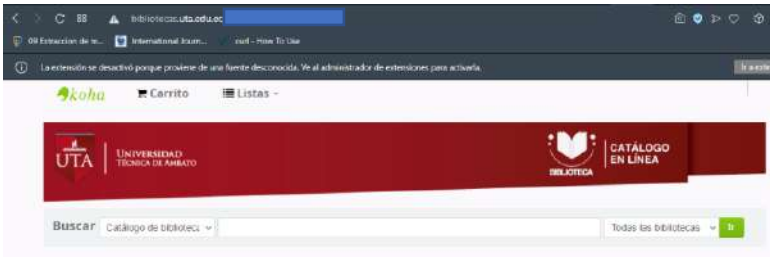



Fuente: elaboración propia

Además, en la Figura 50, se indica la redirección de la página *web* al ingresar a la aplicación, por parte de un determinado usuario.

**Figura 50.**

*Redirección URL Autenticación*




Fuente: elaboración propia

- **Prueba de la funcionalidad de recordar contraseña (OTG-AUTHN-005)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar que ninguna contraseña se almacene en *cookies*, las cuale, logran ser una brecha importante para ser vulnerar por parte de un atacante.

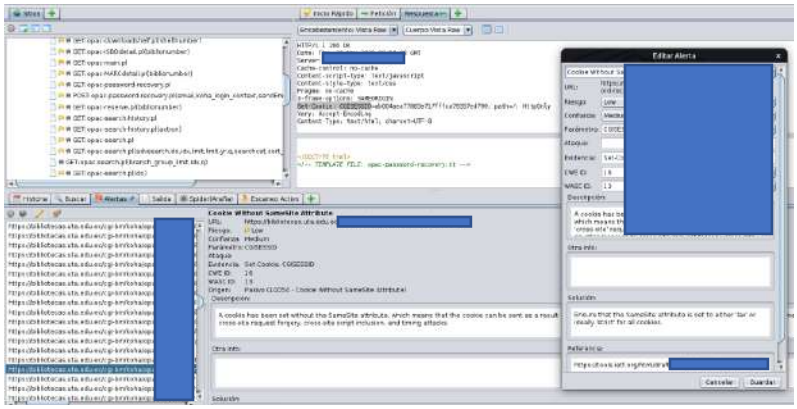
**Ejecución de la Prueba**

Esta prueba, se la realiza mediante la herramienta de OWASP ZAP, para interceptar el acceso de la autenticación y el almacenamiento de *cookies*.

Se manifiesta al ver en la Figura 51, las *cookies* que se almacenan de la aplicación *web*.

**Figura 51.**

*Cookies – Herramienta OWASP Zap*



Fuente: elaboración propia

- **Prueba de la debilidad de la caché del navegador (OTG-AUTHN-006)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar que la aplicación web no guarde en caché del navegador usuario y contraseñas, con ello consigue un atacante hacer uso de ello e interceptar información dentro de la aplicación web.

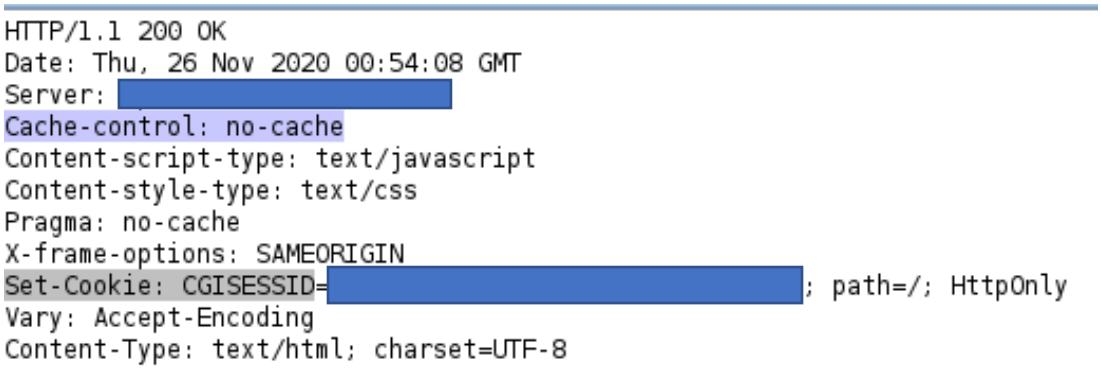
**Ejecución de la Prueba**

Esta prueba, se la realiza mediante la herramienta de OWASP ZAP, para determinar el manejo de *cache*, indicadores adicionales alcanza a ser necesarios para el encabezado Cache-Control y de esta manera prevenir de mejor manera los archivos vinculados persistentemente en el sistema de archivos del servidor *web*.

En la Figura 52, se muestra la evaluación de la página principal de la aplicación *web*, en la que indica que no se almacena la cache.

**Figura 52.**

*Cache del Navegador – Herramienta OWASP Zap*




Fuente: elaboración propia

Se logra verificar el cache relacionado a la página como se muestra en la Figura 53.

**Figura 53.**

*Cache del Navegador – Google*


Fuente: elaboración propia

- **Prueba de la política de contraseñas débiles (OTG-AUTHN-007)**

**Objetivo de la Prueba**

El objetivo de la prueba es determinar la resistencia de la aplicación *web* contra ataques de fuerza bruta y la adivinanza de las contraseñas, para lo cual, se usa diccionarios de contraseñas disponibles en la *web* y que se llevan a cabo mediante la evaluación de los requerimientos como longitud, complejidad, reutilización y caducidad de las contraseñas.

**Ejecución de la Prueba**

Para realizar esta prueba se responderán, a continuación, las siguientes preguntas:

- **¿Qué caracteres son permitidos y prohibidos para usarse en una contraseña? ¿El usuario necesita utilizar caracteres de diferentes conjuntos de caracteres como letras minúsculas y mayúsculas, dígitos y símbolos especiales?**

  Los caracteres son definidos por la lógica de negocio que se presenta al momento de desarrollar una aplicación web, es decir, lo define la Institución o las políticas de usuario y contraseña de esta.

  Si necesita, con ello asegura el usuario, tener una contraseña lo suficientemente segura y difícil de descifran mediante ataques de fuerza bruta.

- **¿Con qué frecuencia logra un usuario cambiar su contraseña? ¿Qué tan rápido logra un usuario cambiar su contraseña después de un cambio anterior? Los usuarios logran eludir requisitos de historial de contraseña cambiando su contraseña cinco veces seguidas para que después del último cambio de contraseña recuperen su contraseña inicial otra vez.**

  Un usuario logra cambiar su contraseña por seguridad cada tres meses.

  Se realizará cuando crea conveniente ero de preferencia seria cada tres meses, pero si maneja información realmente importante lo recomendable seria cada mes.

  Lo ideal sería no repetir contraseñas que ya se hayan utilizado anteriormente para evitar que la contraseña en algún momento se vuelva vulnerable.

- **¿Cuándo un usuario cambiaría su contraseña? ¿Después de 90 días? ¿Después del bloqueo de la cuenta debido a un número excesivo de intentos de inicio de sesión?**

  Cuando la institución le notifique si fuera el caso, o su vez la aplicación podría notificar dicho cambio de contraseña cada cierto tiempo.

  Si seria después de 90 días.

  No necesariamente después de haber olvidado su contraseña.

- **¿Con qué frecuencia consigue un usuario reutilizar una contraseña? ¿La aplicación mantiene un historial de las ultimas ocho contraseñas utilizadas por el usuario?**

  No sería recomendable que un usuario vuelva a reutilizar las contraseñas, logran ser fácil luego para un atacante descubrirla.

  No cuenta con un historial de contraseñas la aplicación web, las mismas son confidenciales.

- **¿Cuán diferente sería la nueva contraseña de la última contraseña usada?**

  Sería totalmente diferente para evitar que sea fácil de romper para un atacante.

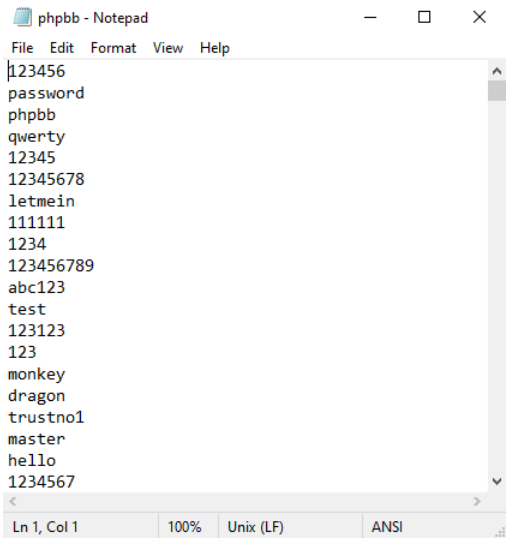
- **¿Se impide al usuario que utilice su nombre de usuario u otra información de la cuenta (como el primer o último nombre) en la contraseña?**

  No cuenta la aplicación web con ese control, los usuario y contraseñas son otorgadas por la institución y se crean con anterioridad

En la Figura 54, se muestra un ejemplo de diccionario de datos para fuerza bruta.

**Figura 54.**

*Diccionario de Datos*




Fuente: elaboración propia

- **Prueba para determinar la seguridad débil de pregunta/respuesta (OTG-AUTHN-008)**

**Objetivo de la Prueba**

El objetivo de la prueba es analizar si la aplicación *web* cuenta con la respectiva seguridad en cuanto a las preguntas que se muestran para recuperar una cuenta o a su vez recuperar contraseñas olvidadas.

Muchas de las ocasiones las preguntas de seguridad se generan con la creación de una cuenta y requieren que el usuario seleccione una o varias preguntas que se encuentran aleatoria o estáticamente generadas y provea una respuesta correcta.

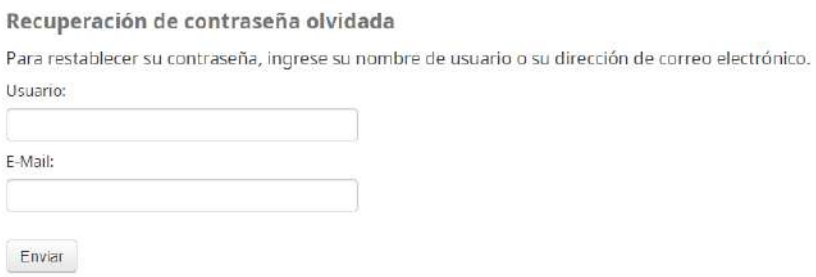
**Ejecución de la Prueba**

No se consigue realizar esta prueba debido a que la recuperación de contraseña, ya que, la aplicación web utiliza el correo institucional.

A continuación, se muestra en la Figura 55, la recuperación de contraseña desde la aplicación.

**Figura 55.**

*Recuperación de Contraseña*




Fuente: elaboración propia

- **Prueba para determinar la seguridad débil de pregunta/respuesta (OTG-AUTHN-009)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar si la aplicación permite a los usuarios cambiar o restablecer rápidamente la contraseña sin que un administrador de la aplicación intervenga.

- Determinar la resistencia de la aplicación web a la subversión del proceso de cambio de la cuenta que permite al usuario cambiar la contraseña de su cuenta.
- Determinar la resistencia de la función de restablecimiento de contraseñas para que no logren eludir o adivinar.

**Ejecución de la Prueba**

No se alcanza a realizar esta prueba debido a que la recuperación de contraseña, la aplicación utiliza de por medio el correo institucional.

- **Prueba de autenticación más débil en canal alternativo (OTG-AUTHN-010)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar si la aplicación permite a los usuarios cambiar o restablecer su cuenta y contraseña mediante otros medios, es decir, dispositivos móviles, o a su vez a través de un Centro de llamadas.

**Ejecución de la Prueba**

Para realizar esta prueba, a continuación, se detalla los medios por los cuales el usuario logra hacer el uso de la función para restablecer la contraseña.

En la Tabla 7, se muestra los diferentes medios por los cuales el usuario consigue reestablecer la contraseña de la cuenta.

**Tabla 7.**

*Canales de Autenticación*

| Funcionalidad | Dispositivo Móvil | Centro de llamadas | Sitio *Web* |
|---|---|---|---|
| Registro | Si | No | Si |
| Inicio de sesión | Si | No | Si |
| Cerrar sesión | Si | No | Si |
| Reestablecer contraseña | Si | No | Si |

Fuente: elaboración propia

### 2.3.5 Pruebas de Autorización

La autorización, es el concepto de permitir que el acceso a los recursos solo a aquellos usuarios a quienes se les otorga el permiso para usarlos. El probar la autorización, significa comprender cómo funciona el proceso de autorización y utilizar dicha información para eludir algún mecanismo de autorización.

La autorización, es un proceso que viene después de una autenticación exitosa, por lo que el evaluador verificará este punto después de que tenga credenciales válidas, asociadas con un conjunto bien definido de roles y privilegios. Durante este tipo de evaluación, se verificará si es posible omitir el esquema de autorización, encontrar una vulnerabilidad de recorrido de ruta o encontrar formas de escalar los privilegios asignados al evaluador.

**OTG-AUTHZ-001**

**OTG:** OWASP TEST GUIDE v4.0 (Guía de Pruebas de OWASP v4.0)

**AUTHZ:** Se refiere a la categoría de cada fase, en este caso (Pruebas de Autorización)

**001:** Se refiere al número de categoría a la que pertenece, para el caso es: prueba número uno de la categoría pruebas de autorización.

- **Prueba de Inclusión de archivos/recorrido de directorio (OTG-AUTHZ-001)**

**Objetivo de la Prueba**

El objetivo de la prueba es conocer si existe o no suficiente seguridad en los diferentes procesos de autorización para usuarios, es decir, el usuario alcanza a acceder a cualquier carpeta desde la raíz de la aplicación *web*.

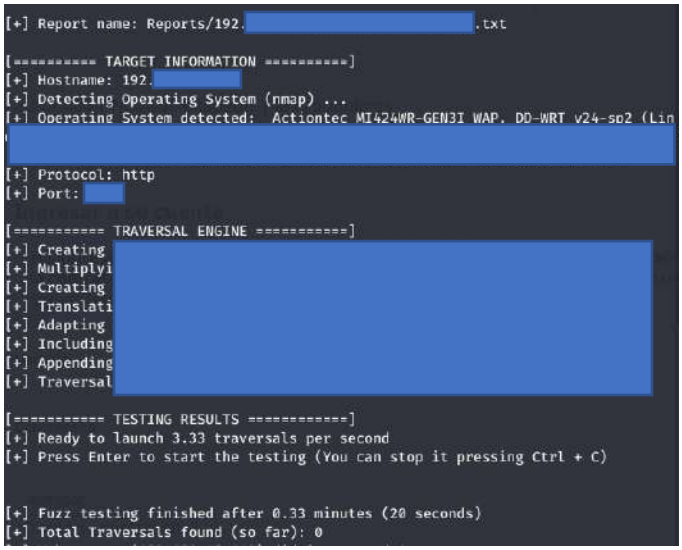
**Ejecución de la Prueba**

Para realizar la comprobación, se utiliza la herramienta DotDotpwn, la cual, analizará los directorios transversales en la aplicación *web*.

A continuación, en la Figura 56, se identifica si esta la seguridad respectiva para la autorización de permisos de directorios, es decir, un análisis transversal, los cuales están en ejecución.

**Figura 56.**

*Path Transversal – Herramienta DotDotpwn*




Fuente: elaboración propia

- **Prueba para eludir el esquema de autorización (OTG-AUTHZ-002)**

**Objetivo de la Prueba**

El objetivo de la prueba es comprobar, cómo se implementó el esquema de autorización para que cada rol o privilegio obtenga acceso a funciones reservadas y recursos de la aplicación *web*.

**Como comprobar**

No se consigue realizar esta prueba debido a que no se logra obtener acceso a todos los directorios y los diferentes roles de la aplicación *web*.

- **Prueba de escalamiento de privilegios (OTG-AUTHZ-003)**

**Objetivo de la Prueba**

El objetivo de la prueba es identificar si a un atacante mediante esta vulnerabilidad le permite ganar privilegios dentro de la aplicación *web*, y poder realizar más actividades de las que logra realizar cualquier usuario dentro del sistema.

Generalmente en sistemas operativos *Windows* un atacante intenta lograr tener permisos de administrador, mientras que para usuarios de sistemas *Linux* intentan tener privilegios de *root*.

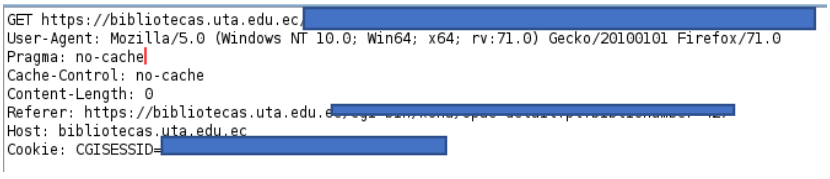
**Ejecución de la Prueba**

Para realizar esta prueba, se utiliza la herramienta OWASP Zap para interceptar las *urls* que envían datos informativos a través de ellos.

En la Figura 57, se muestra el id de un usuario autenticado que se envía a través de la *url* de la aplicación *web*.

**Figura 57.**

*Session ID – OWASP Zap*




Fuente: elaboración propia

- **Prueba de referencias de objetos directos inseguros (OTG-AUTHZ-004)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar como un atacante consiga modificar de manera fácil los parámetros establecidos por la aplicación. Cuando dicho ataque es exitoso el atacante logra sobrepasar los permisos de autorización y así acceder a recurso dentro del sistema.

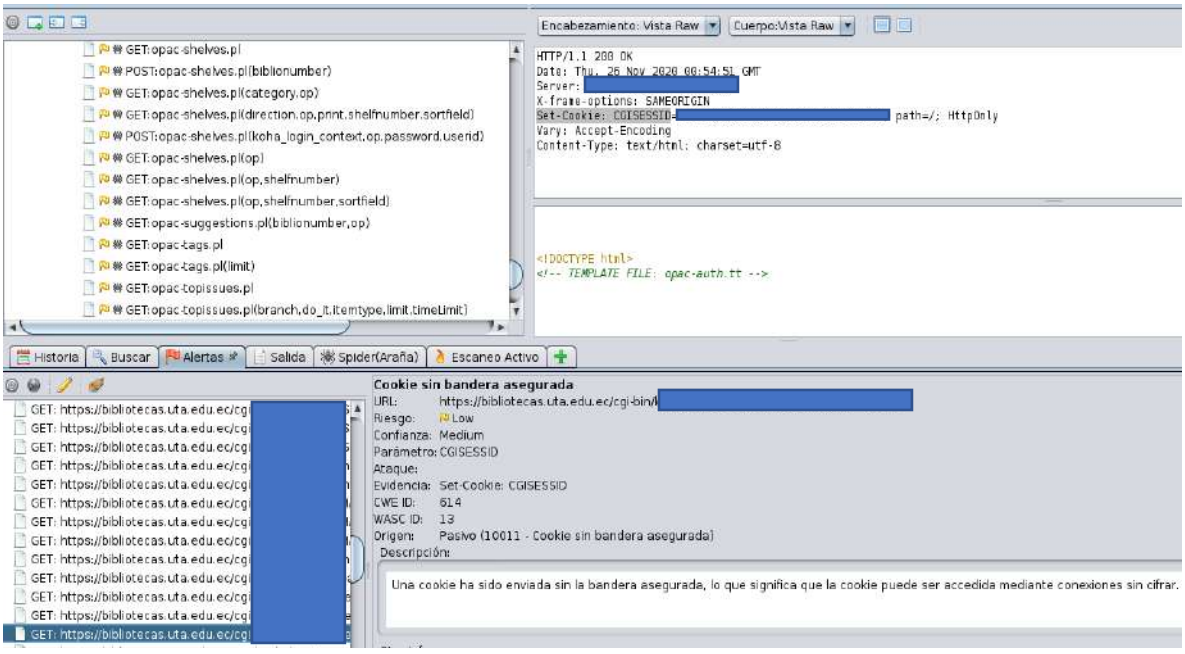
**Ejecución de la Prueba**

Para realizar esta prueba, se utilizará la herramienta OWASP Zap, la cual, ayuda a determinar las *URLS* que envían datos mediante los métodos *GET* y *POST*, valores de sesiones que alcancen a ser confidenciales en muchos de los casos, las cuales un atacante logre hacer uso de ellas para logran algún cometido.

A continuación, en la Figura 58, se muestra los valores que se envía a través de los métodos *GET* y *POST*, las cuales logran ser fácilmente interceptadas y modificadas, como lo indica, además, en el mensaje de la herramienta.

**Figura 58.**

*URL Interceptada – OWASP Zap*




Fuente: elaboración propia

### 2.3.6 Pruebas de Gestión de Sesiones

Uno de los componentes centrales de cualquier aplicación basada en *web* es el mecanismo mediante, el cual, controla y mantiene el estado de un usuario que interactúa con ella. Esto se conoce como administración de sesiones y se define como el conjunto de todos los controles que gobiernan la interacción de estado completo entre un usuario y la aplicación basada en la *web*. Esto cubre ampliamente cualquier cosa, desde, realizar la autenticación del usuario, hasta, qué sucede cuando se desconecta.

**OTG-SESS-001**

**OTG:** OWASP TEST GUIDE v4.0 (Guía de Pruebas de OWASP v4.0)
**SESS:** Se refiere a la categoría de cada fase, en este caso (Pruebas de Gestión de Sesión)
**001:** Se refiere al número de categoría a la que pertenece, para el caso es: prueba número uno de la categoría pruebas de gestión de sesión.

- **Prueba para omitir el esquema de administración de sesiones (OTG-SESS-001)**

**Objetivo de la Prueba**

El objetivo de la prueba es conocer la configuración para manejo de sesiones, dentro de la aplicación *web*, además, se comprobaría que las *cookies* y otras fichas de sesión se crean de manera segura e impredecible, un atacante es capaz de predecir y falsificar una *cookie*, para así secuestra la sesión de usuarios que se encuentran legitimante conectados, los atacantes generalmente realizan los siguientes pasos para obtener información en cuanto a las *cookies* se refiere:

- **Recolección de *cookies***
- **Ingeniería Inversa de *cookies***
- **Manipulación de *cookies***

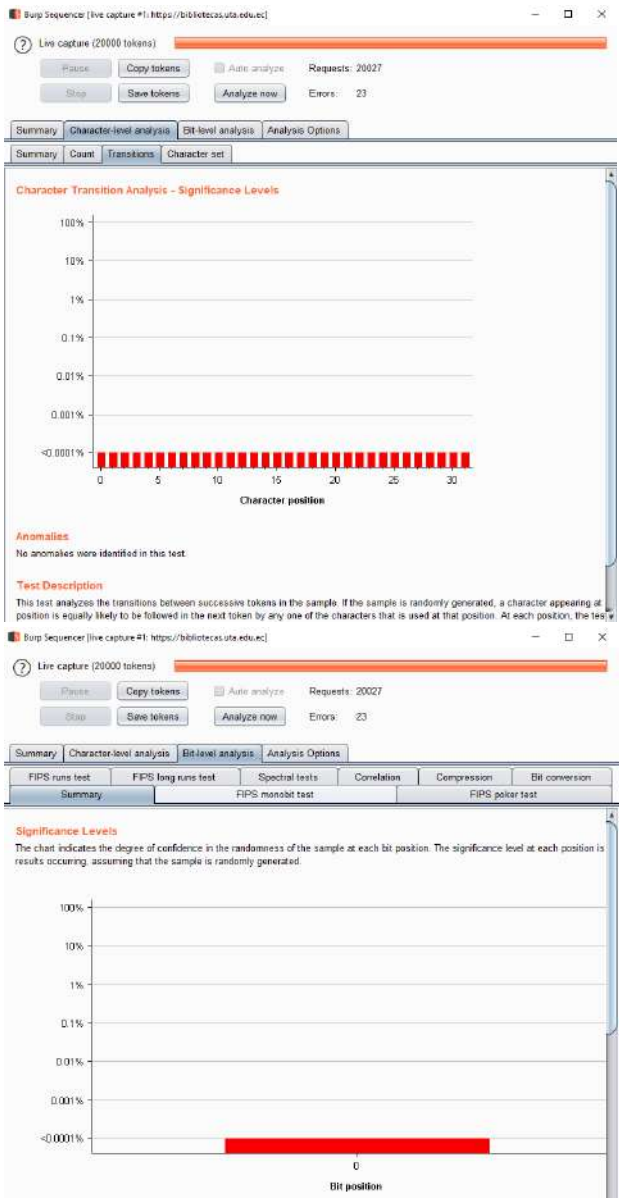
**Ejecución de la Prueba**

Para realizar la identificación de esta configuración, se utiliza la herramienta BurpSuite, la cual, permite capturar las *cookies* que se ejecutan dentro de la aplicación.

A continuación, en la Figura 59, se muestra las *cookies* capturadas desde la aplicación *web*.

**Figura 59.**

*Análisis de Cookies – Herramienta Burp Suite*




Fuente: elaboración propia

- **Prueba de fijación de sesión (OTG-SESS-003)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar cuando una aplicación no renueva las *cookies* pertenecientes a la aplicación y respectivamente a la sesión después de una autenticación de usuario exitosa, es posible encontrar una vulnerabilidad de fijación de sesión y de alguna forma obligar al usuario a utilizar una *cookie* conocida por el atacante. De ser ese el caso, un atacante podría robar la sesión del usuario convirtiéndose ello en un secuestro de sesión.

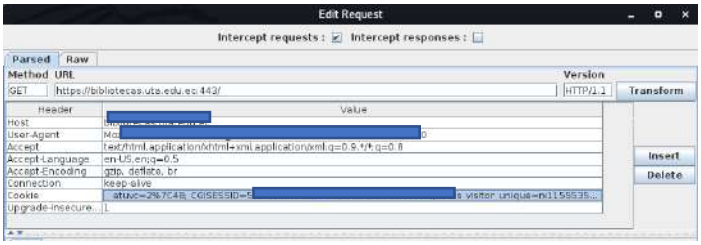
**Ejecución de la Prueba**

Para realizar la identificación de *cookies*, y la intercepción de estas, se utiliza la herramienta WebScarab.

A continuación, en la Figura 60, se muestra la intercepción de *cookies*, para realizar un ataque, la cual, determina el Host, el Agente, y la respectiva cookie que maneja la aplicación web.

**Figura 60.**

*Intercepción de Cookies – Herramienta WebScarab*




Fuente: elaboración propia

- **Prueba de falsificación de solicitudes entre sitios CSRF (OTG-SESS-005)**

**Objetivo de la Prueba**

Los tokens de sesión (*cookie*, ID de sesión, campo oculto), si están expuestos normalmente permitirán a un atacante hacerse pasar por una víctima y acceder a la aplicación de forma ilegítima. Es importante que estén protegidos contra escuchas en todo momento, especialmente mientras se encuentran en tránsito entre el navegador del cliente y los servidores de aplicaciones.

**Ejecución de la Prueba**

Para realizar la identificación CRSF, se utiliza la herramienta OWASP Zap, y realizar una prueba de interceptar esta vulnerabilidad.

A continuación, en la Figura 61, se muestra la intercepción de un Cross-site request forgery (CRSF) en la aplicación *web*.

**Figura 61.**

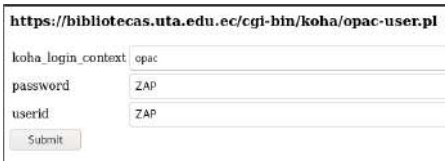
*Intercepción de CRSF – Herramienta OWASP Zap*

Fuente: elaboración propia

También, se muestra un ejemplo de interceptar CRSF, en la Figura 62.

**Figura 62.**

*Página de CRSF – Herramienta OWASP Zap*




Fuente: elaboración propia

- **Prueba de la funcionalidad de cierre de sesión (OTG-SESS-006)**

**Objetivo de la Prueba**

La terminación de la sesión es una parte importante del ciclo de vida de la sesión. Reducir al mínimo la vida útil de los tokens de sesión disminuye la probabilidad de un ataque de secuestro de sesión exitoso. Esto logra verse como un control para evitar otros ataques como *Cross Site Scripting y Cross Site Request Forgery*. Se sabe que dichos ataques dependen de que un usuario tenga una sesión autenticada presente. No tener una terminación de sesión segura solo aumenta la superficie de ataque para cualquiera de estos ataques.

Una terminación de sesión segura requiere al menos los siguientes componentes:

- Disponibilidad de controles de interfaz de usuario que permiten al usuario cerrar sesión manualmente.
- Terminación de la sesión después de un período de tiempo determinado sin actividad (tiempo de espera de la sesión).
- Invalidación adecuada del estado de la sesión del lado del servidor.

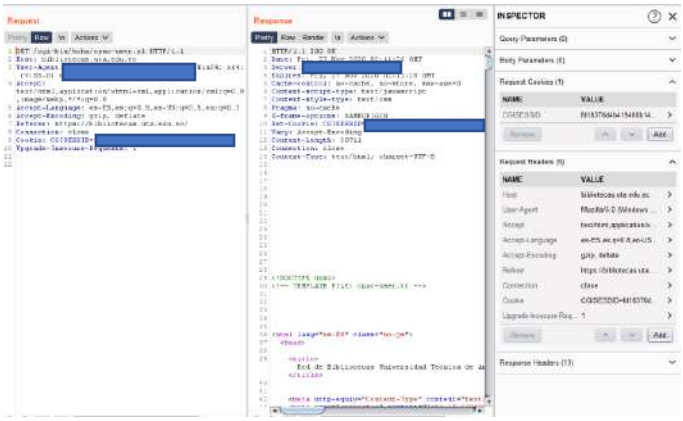
**Ejecución de la Prueba**

Para realizar la verificación de la terminación de sesión, se utiliza la herramienta Burp Suite, la cual, ayuda a verificar esta vulnerabilidad que es muy común en las aplicaciones es *web*.

A continuación, en la Figura 63 se muestra la finalización de la sesión en la aplicación *web*, con esto se podría fácilmente interceptar la sesión y enviar un cierre imprevisto.

**Figura 63.**

*Finalización de sesión – Herramienta Burp Suite*




Fuente: elaboración propia

### 2.3.7 Pruebas de Validación de Entrada

La debilidad de seguridad de las aplicaciones web más común es la falla en validar adecuadamente la entrada proveniente del cliente o del entorno antes de usarla. Esta debilidad conduce a casi todas las principales vulnerabilidades en las aplicaciones *web*, como secuencias de comandos entre sitios, inyección SQL *(Structured Query Language)*, inyección de intérprete, ataques *locale / Unicode*, ataques al sistema de archivos y desbordamientos de búfer.

**OTG-INPVAL-001**

**OTG:** OWASP TEST GUIDE v4.0 (Guía de Pruebas de OWASP v4.0)

**INPVAL:** Se refiere a la categoría de cada fase, en este caso (Prueba de Validación de Entrada)

**001:** Se refiere al número de categoría a la que pertenece, para el caso es: prueba número uno de la categoría pruebas de validación de entrada.

- **Prueba de secuencia de comandos de sitios cruzados reflejados (OTG-INPVAL-001)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar la correcta codificación de los caracteres, en muchos de los casos los servidores no filtran algunas codificaciones de caracteres, esta vulnerabilidad mediante XSS, la cual, un atacante utiliza una aplicación web para de esa manera enviar algún código malicioso, generalmente en forma de un script del lado del navegador, a un usuario final diferente.

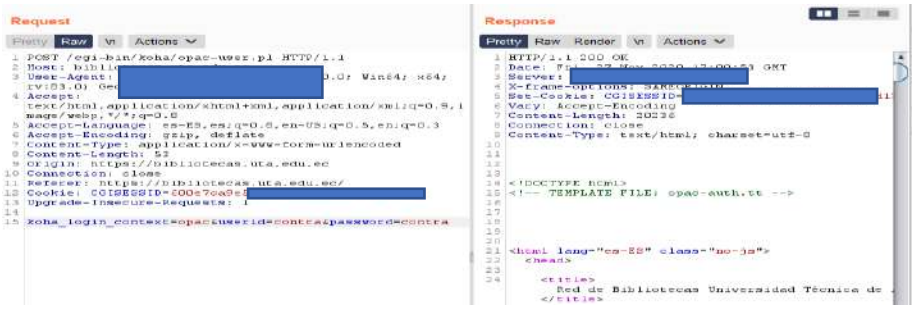
**Ejecución de la Prueba**

Para realizar la identificación de esta funcionalidad, se obtiene un detalle de la vulnerabilidad de XSS, la cual, se lo realiza mediante la herramienta Burp Suite, la cual, intercepta esta característica y se podrá ver de la mejor manera, esto se verifica mediante el método *GET* o *POST*.

A continuación, en la Figura 64, se muestra la intercepción de una contraseña de un usuario dentro de la aplicación *web*.

**Figura 64.**

*XSS – Herramienta Burp Suite*




Fuente: elaboración propia

- **Prueba de manipulación verbos (OTG-INPVAL-003)**

**Objetivo de la Prueba**

El objetivo de la prueba determinar posee métodos alternativos a *GET* Y *POST* para el envío y transmisión de información, estos métodos alternativos de igual forma responden a solicitudes desde la aplicación *web* de una manera no prevista por los desarrolladores, cuando estos son puestos en marcha.

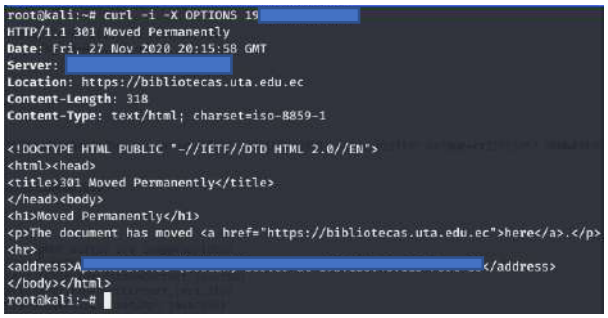
**Ejecución de la Prueba**

Para realizar la prueba, se realizará mediante la herramienta curl, la solicitud de los métodos HTTP.

A continuación, en la Figura 65, se muestra los métodos que se ejecutan en el servidor, pero para el caso de la aplicación *web*, no se muestra, tiene configurado para no mostrar esa actividad.

**Figura 65.**

*Métodos HTTP – Herramienta Curl*




Fuente: elaboración propia

- **Prueba de contaminación de parámetros HTTP (OTG-INPVAL-004)**

**Objetivo de la Prueba**

El objetivo de la prueba determinar si existe alguna manera de realizar un ataque de contaminación de parámetros que afecten al comportamiento de los formularios del lado del cliente, es decir, por medio de parámetros quienes se encargan de enviar dichos datos a la aplicación *web* mediante *urls*, para que esta funcione aparentemente de manera adecuada, generalmente, se ve en formularios de ingreso de usuarios, para enviar correos electrónicos, para comentar, llenar encuestas, realizar búsquedas de información entre otros.

**Ejecución de la Prueba**

Para realizar la prueba, se realizará mediante la herramienta OWAZP Zap, para ver la contaminación de parámetros mediante el protocolo HTTP, esta contaminación de parámetros, se denomina HPP *(High Pressure Processing)*.

A continuación, se muestra la Figura 66, la cual, determina los parámetros que mediante la *url* se pasan para realizar determinadas búsquedas dentro de la aplicación *web*.

**Figura 66.**

*HPP – Herramienta OWASP Zap*


Fuente: elaboración propia

- **Prueba de inyección SQL (OTG-INPVAL-005)**

**Objetivo de la Prueba**

El objetivo de la prueba es verificar si existe la manera de realizar el ataque de inyección SQL, mediante este ataque a la aplicación *web,* se alcanza a leer datos confidenciales de una base de datos, además, modificar los datos de la base de datos mediante los procesos de (insertar / actualizar / eliminar), ejecutar operaciones administrativas en la base de datos como, por ejemplo, cerrar el DBMS *(Database Management System)*, entre otras operaciones sobre la base de datos.

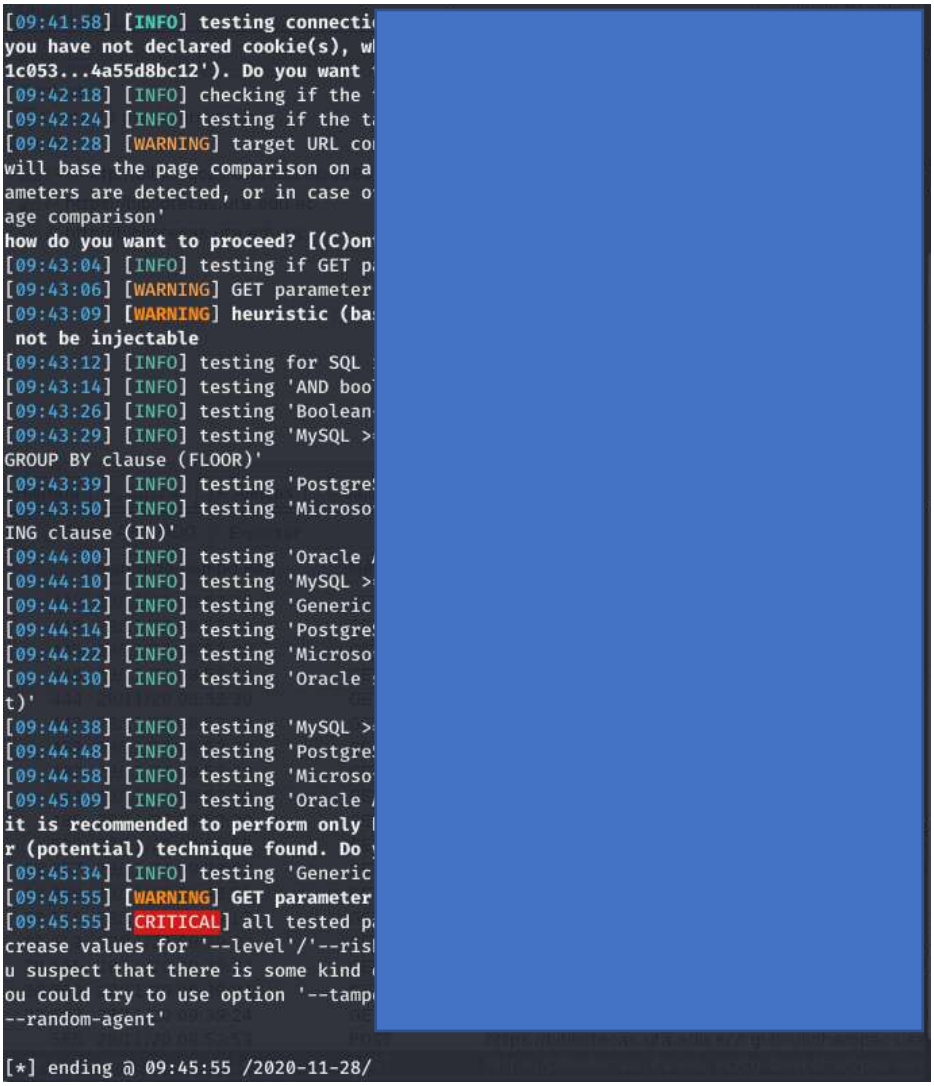
**Ejecución de la Prueba**

Para realizar la prueba, se realizará mediante la herramienta sqlmap, la cual, ayudará a verificar si existe alguna vulnerabilidad de inyección de SQL.

A continuación, en la Figura 67, se muestra el análisis de vulnerabilidades de Inyección SQL.

**Figura 67.**

*Inyección SQL – Herramienta SqlMap*




Fuente: elaboración propia

- **Prueba de inyección de comando (OTG-INPVAL-013)**

**Objetivo de la Prueba**

El objetivo de la prueba es comprobar una aplicación para la inyección de comandos del sistema operativo, esta prueba intenta inyectar un comando del sistema operativo web a través de una solicitud HTTP a la aplicación.

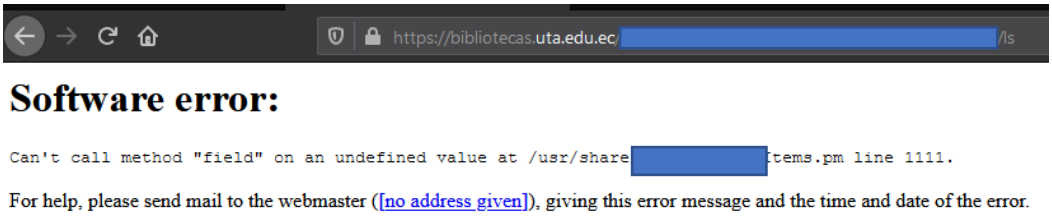
**Ejecución de la Prueba**

Para realizar la prueba, se realizará de manera manual en la *url*, inyectando código al sistema operativo mediante una solicitud HTTP.

A continuación, en la Figura 68, se muestra el comando inyectando el comando para obtener información del sistema operativo.

**Figura 68.**

*Inyección de comando*




Fuente: elaboración propia

### 2.3.8 Pruebas de Manejo de Errores

El manejo de errores en las aplicaciones web, es muy indispensable, al controlar cada uno de ellos mediante algún proceso a prueba de fallos, hace que a aplicación sea más rápida y eficiente, es por ello por lo que las aplicaciones hoy en día necesitan de un control exhaustivo ante este tipo de problemas, que para un ciberdelincuente consiga ser un paso importante para vulnerar una aplicación *web*.

**OTG-ERR-001**

**OTG:** OWASP TEST GUIDE v4.0 (Guía de Pruebas de OWASP v4.0)

**ERR:** Se refiere a la categoría de cada fase, en este caso (Prueba de Manejo de Errores)

**001:** Se refiere al número de categoría a la que pertenece, para el caso es: prueba número uno de la categoría pruebas de manejo de errores.

- **Prueba análisis de códigos de error (OTG-ERR-001)**

**Objetivo de la Prueba**

El objetivo de la prueba es determinar si una página de la aplicación *web*, contiene algún control a prueba de errores, ya sean estas por conexión o parámetros enviados de manera incorrecta, o algún problema en cuanto a respuesta desde el servidor *web*.

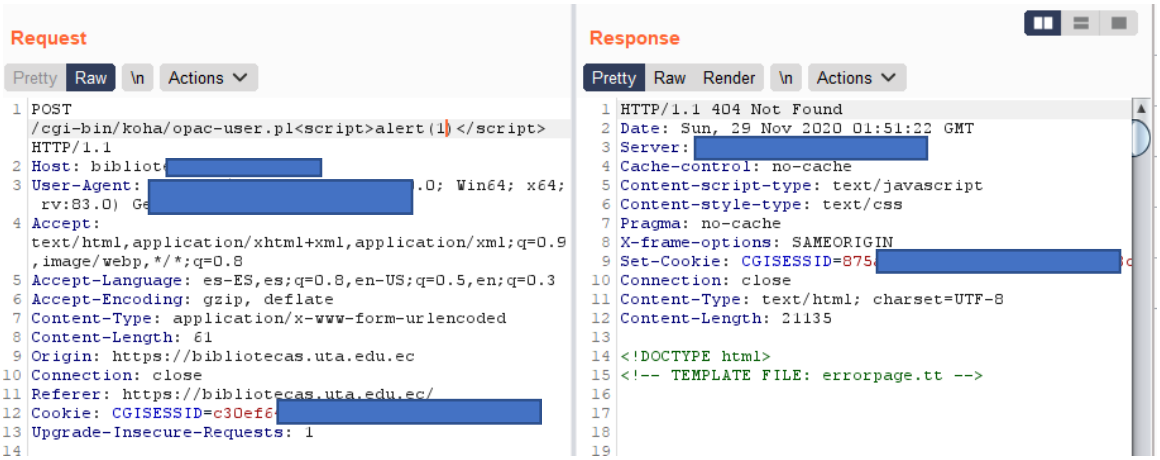
**Ejecución de la Prueba**

Para realizar la identificación de esta funcionalidad, se va a enviar a través de la *url* de una página web de la aplicación un script para determinar si existe un manejo de errores, utiliza la herramienta Burp Suite.

A continuación, en la Figura 69, se muestra el error de respuesta HTTP 404, que emite la aplicación.

**Figura 69.**

*Error HTTP 404 – Herramienta Burp Suite*

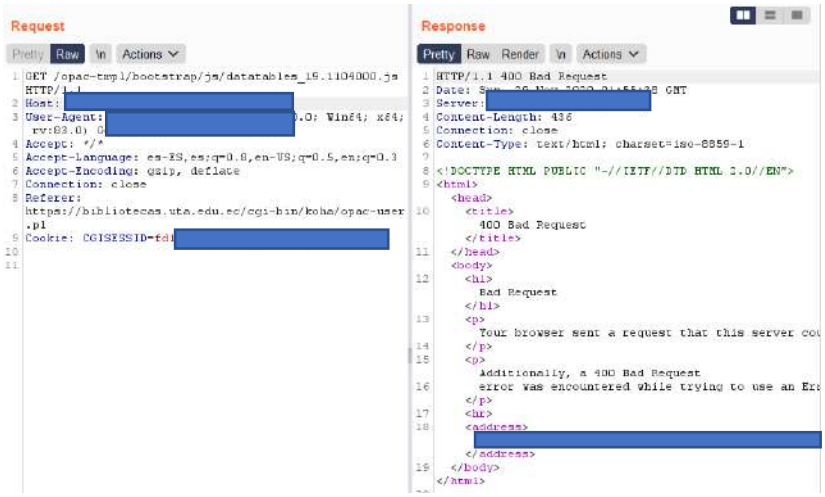



Fuente: elaboración propia

También, se ve en la Figura 70 el error de respuesta HTTP 400, que emite la aplicación luego de un error.

**Figura 70.**

*Error HTTP 400 – Herramienta Burp Suite*




Fuente: elaboración propia

En la Figura 71, se muestra el error de respuesta HTTP 404, a través del navegador desde la aplicación.

**Figura 71.**

*Error HTTP 404*


Fuente: elaboración propia

### 2.3.9 Pruebas de Criptografía débil

**OTG-CRYPST-001**

**OTG:** OWASP TEST GUIDE v4.0 (Guía de Pruebas de OWASP v4.0)

**CRYPST:** Se refiere a la categoría de cada fase, en este caso (Prueba de Criptografía débil)

**001:** Se refiere al número de categoría a la que pertenece, para el caso es: prueba número uno de la categoría pruebas de criptografía débil.

- **Prueba de cifrados SSL / TLS débiles, protección insuficiente de la capa de transporte (OTG-CRYPST-001)**

**Objetivo de la Prueba**

El objetivo de la prueba es determinar si la aplicación web, contiene un certificado TLS/SSL *(Transport Layer Security) / (Secure Sockets Layer)*, la cual, ayuda a mantener segura la aplicación web, además, se verifica si la información que se envía entre el usuario y la aplicación web, para que estas viajen de manera segura.

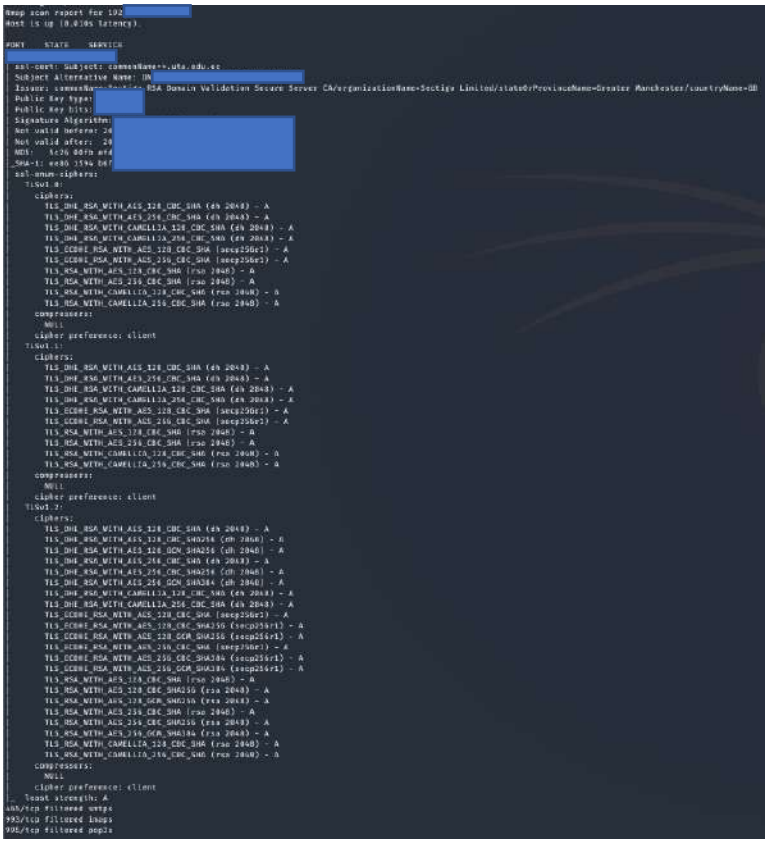
**Ejecución de la Prueba**

Para realizar la prueba, se realiza un escaneo mediante la herramienta *nmap* para determinar si está configurado SSL y está utilizada para evitar la puesta en marcha del soporte criptográfico que podría ser fácilmente derrotado.

A continuación, en la Figura 72, se muestra el resultado de los certificados TLS/SSL, mediante la herramienta *nmap*.

**Figura 72.**

*TLS/SSL – Herramienta Nmap*

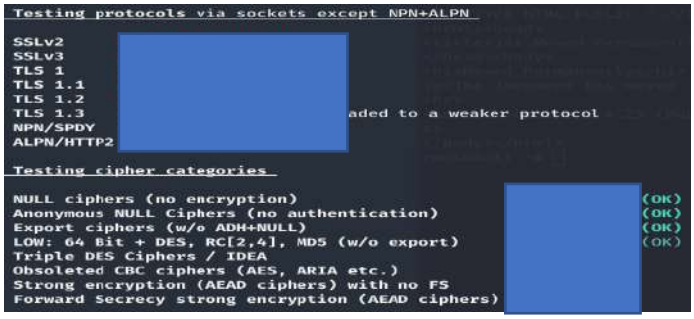


Fuente: elaboración propia

En la Figura 73, se muestra los detalles de los certificados SSL, en la aplicación *web*, mediante la herramienta testssl.sh

**Figura 73.**

*TLS/SSL – Herramienta testssl.sh*




Fuente: elaboración propia

- **Prueba de información confidencial enviada a través de canales no cifrados (OTG-CRYPST-003)**

**Objetivo de la Prueba**

El objetivo de la prueba es determinar si los datos que se envía a través de la red son cifrados, o, si los datos que se envían, se los realiza mediante HTTPS *(Hypertext Transfer Protocol Secure)*, este mecanismo de protección no tendría limitaciones o vulnerabilidades.

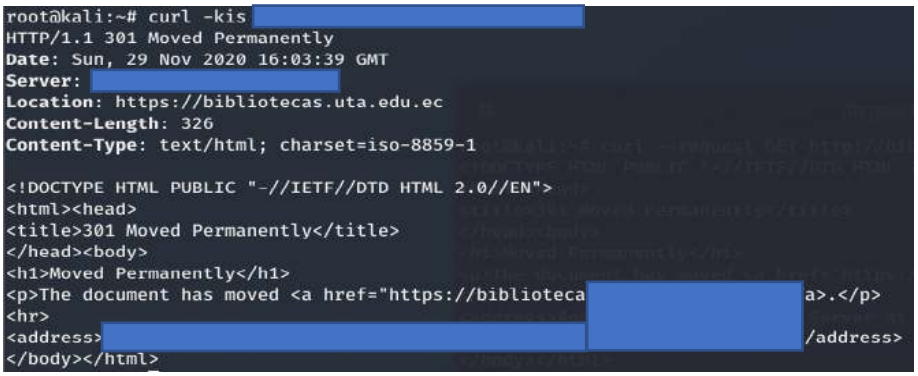
**Ejecución de la Prueba**

Para realizar la prueba, se utiliza la herramienta *curl*, para verificar el uso de autenticación básica sobre HTTP. Cuando se utiliza la autenticación básica, las credenciales de un determinado usuario, se codifican en lugar de cifrarse y se envían como encabezados del protocolo HTTP.

A continuación, en la Figura 74, se logra verificar la codificación de la aplicación web.

**Figura 74.**

*Codificación Sitio Web– Herramienta curl*




Fuente: elaboración propia

# CAPÍTULO III. ANALISIS DE LOS RESULTADOS DE LA INVESTIGACIÓN

En este capítulo, se detalla los resultados obtenidos al realizar las pruebas que determina la metodología OWASP para las aplicaciones web, se basa en la Guía de Pruebas de OWASP v4.0, además, se realiza el análisis de dichos resultados, con la finalidad de comprobar si existen o no vulnerabilidades en las aplicaciones web de la Institución.

## 3.1 Resumen de las pruebas OWASP

A continuación, se realiza un resumen de las pruebas realizadas a una aplicación web de la Institución.

### 1.1.1 Recopilación de información

Resumen de las pruebas de recopilación de información realizadas a la aplicación web de la Institución.

La Tabla 8 muestra una lista de resumen, de las pruebas realizadas, con el respectivo objetivo de cada una de ellas y las herramientas utilizadas en su respectiva implementación para el análisis.

**Tabla 8.**

*Recopilación de información*

| CATEGORÍA | CÓDIGO DE PRUEBA | NOMBRE DE LA PRUEBA | OBJETIVO DE LA PRUEBA | HERRAMIENTAS UTILIZADAS |
|---|---|---|---|---|
| Recopilación de la Información | OTG-INFO-001 | Realizar el descubrimiento y reconocimiento del motor de búsqueda para la fuga de información | El objetivo de la prueba es buscar la mayor cantidad de información en el Internet, acerca del sitio web que se analiza y, además, verificar si existe algún tipo de información expuesta. | Google, Shodan, Google Dorks |

| | | | | |
|---|---|---|---|---|
| | OTG-INFO-002 | Servidor web de huellas digitales (OTG-INFO-002) | El objetivo de la prueba es encontrar la versión y a su vez el tipo de servidor que se aloja la aplicación web que es el objetivo, en las cuales se podría encontrar vulnerabilidades conocidas y la manera de explotarlas durante las pruebas. | Curl, Gregthatcher, Httprecon |
| | OTG-INFO-003 | Revisar metarchivos del servidor web para detectar fugas de información (OTG-INFO-003) | El objetivo de la prueba es buscar fugas de información de la ruta o rutas al directorio o carpeta de la aplicación web, que se lo realiza mediante el uso de la información del archivo robots.txt. Además, se alcanza a crear lista de archivos de directorios los cuales no son *indexados* por arañas, *robots* o rastreadores. | Wget, Burp Suite |
| | OTG-INFO-004 | Enumerar aplicaciones en el servidor web (OTG-INFO-004) | El objetivo de la prueba es identificar las aplicaciones que se ejecutan en el servidor web, muchas de las cuales tiene vulnerabilidades y estrategias de ataques conocidas que logran ser explotadas y de esta manera obtener datos importantes de la aplicación web, además, se logra obtener información del sistema operativo. | Nmap, DNSStuff, NetCraft |

| | | | | |
|---|---|---|---|---|
| | OTG-INFO-005 | Revisión de los comentarios y metadatos de la página web para detectar fugas de información (OTG-INFO-005) | El objetivo de la prueba es revisar los comentarios y metadatos de la aplicación web, que ayudara a entender mejor la aplicación y así encontrar algún tipo de fuga de información. Se logra decir que es muy común e incluso recomendable para programadores incluir comentarios detallados y metadatos en el código fuente de una determinada aplicación *web*. Pero por tal situación dichos comentarios y metadatos al ser incluidos en el código HTML estos podrían revelar información importante. | Google, Curl |
| | OTG-INFO-006 | Identificar los puntos de entrada de la aplicación (OTG-INFO-006) | El objetivo de la prueba es entender cómo se forman las solicitudes y las respuestas que se dan mediante la aplicación *web*, que identifica los métodos *GET* y *POST* respectivamente, sus respectivas variables que intervienen en el proceso de intercambio de datos, además, permiten al evaluador identificar probables áreas de debilidad. Ayuda a identificar y mapear las áreas que se encuentran dentro de la aplicación web que se investigarían una vez que la enumeración y el mapeo se complete. | Burp Suite, OWASP ZAP |

| | | | | |
|---|---|---|---|---|
| | OTG-INFO-007 | Mapa de rutas de ejecución a través de la aplicación (OTG-INFO-007) | El objetivo de la prueba es el crear mapas de la aplicación web de destino y así comprender los principales flujos de trabajo que este realiza. | Burp Suite, OWASP ZAP |
| | OTG-INFO-008 | Marco de aplicación web de huellas dactilares (OTG-INFO-008) | El objetivo de la prueba es el definir un *framework* que utiliza la aplicación web para la búsqueda de vulnerabilidades, se utiliza varios proveedores y versiones de *frameworks web*, lo que hacen la mayoría de ellos es buscar un marcador desde una ubicación preestablecida y luego compararlo con la base de datos de dichas firmas conocidas. | WhatWeb, Wappalyzer |
| | OTG-INFO-009 | Aplicación web de huellas dactilares (OTG-INFO-009) | El objetivo de la prueba es identificar la aplicación *web* y a su vez la versión, para determinar algunas vulnerabilidades conocidas, la forma apropiada para explotarlas durante la prueba. | BlindElephant, Wappalyzer |

Fuente: elaboración propia

### 1.1.2 Pruebas de gestión de la configuración y la implementación

Resumen de las pruebas de gestión de la configuración y la implementación realizadas a la aplicación web de la Institución.

La Tabla 9 muestra una lista de resumen de las pruebas realizadas, con el respectivo objetivo de cada una de ellas y las herramientas utilizadas en su respectiva implementación para el análisis.

**Tabla 9.**

*Pruebas de gestión de la configuración y la implementación*

| CATEGORÍA | CÓDIGO DE PRUEBA | NOMBRE DE LA PRUEBA | OBJETIVO DE LA PRUEBA | HERRAMIENTAS UTILIZADAS |
|---|---|---|---|---|
| Pruebas de gestión de la configuración y la implementación | OTG-CONFIG-001 | Prueba de configuración de red / infraestructura (OTG-CONFIG-001) | El objetivo de la prueba es conocer la infraestructura de red, conocer como están conectados los diferentes dispositivos, servidores, computadoras, firewall. | - |
| | OTG-CONFIG-002 | Configuración de la plataforma de aplicaciones de prueba (OTG-CONFIG-002) | El objetivo de la prueba es conocer si las instalaciones que se han realizado dentro de la aplicación fueron de manera manual o si se han realizado instalaciones automáticas. | Apache2ctl |
| | OTG-CONFIG-003 | Manejo de extensiones de archivo de prueba para información confidencial (OTG-CONFIG-003) | El objetivo de la prueba es verificar como los servidores manejan las peticiones a las diferentes extensiones, ayuda a comprender mejor el comportamiento del servidor. | Nikto |
| | OTG-CONFIG-004 | Revisar archivos antiguos, de copia de seguridad y sin referencia en busca de información confidencial | El objetivo de la prueba es, verificar archivos viejos que no se encuentren referenciados, los cuales contienen datos sensibles, además, existen archivos que se crean como consecuencia de editar otros archivos, los cuales contienen la | Screaming Frog SEO Spider |

| | | | | |
|---|---|---|---|---|
| | | (OTG-CONFIG-004) | misma información que el archivo original. | |
| | OTG-CONFIG-005 | Infraestructura de enumeración e interfaces de Administración de Aplicaciones (OTG-CONFIG-005) | El objetivo de la prueba es encontrar interfaces de administrador, las cuales logran estar presentes directamente en la aplicación o en el servidor donde se aloja la aplicación, mediante estas interfaces permiten al usuario que tienen privilegios hacer realizar actividades que los usuarios no autorizados no los consiguen realizar. | - |
| | OTG-CONFIG-006 | Prueba de Métodos HTTP (OTG-CONFIG-006) | El objetivo de la prueba identificar los métodos que la aplicación web maneja en el servidor, HTTP (Hypertext Transfer Protocol) maneja métodos de GET y POST, estos serían configurados de manera correcta, pues alcanzan a ser estos métodos utilizados para fines delictivos por parte de los ciberdelincuentes, algunos de estos métodos se consiguen plantear un potencial riesgo para la aplicación web, y para las demás aplicaciones que se manejan dentro de la Institución. | Nmap |

| | | | | |
|---|---|---|---|---|
| | OTG-CONFIG-007 | Prueba de Seguridad de Transporte Estricto HTTP - HSTS (OTG-CONFIG-007) | El objetivo de la prueba es probar si existe la presencia del encabezado HSTS (Strict Transport Security) que es un mecanismo de política de seguridad web, el cual, ayuda a proteger los sitios o aplicaciones web contra ataques informáticos, por ejemplo, ataques de degradación de protocolo y secuestro de cookies, para lo cual, se alcanza a comprobar la existencia de la Rubrica HSTS en la respuesta del servidor de un proxy, o a su vez utiliza el comando curl. | Curl, hstspreload, Qualys SSL Labs |
| | OTG-CONFIG-008 | Prueba de Política de Dominio cruzado RIA (OTG-CONFIG-008) | Ria (Rich Internet Applications), son aplicaciones basadas en la Web que tienen algunas características de las aplicaciones gráficas de escritorio, adopto las características de políticas de crossdomain.xml, esto quiere decir que un dominio logra conceder acceso remoto a los servidores desde otro dominio totalmente diferente. | Nikto |

Fuente: elaboración propia

### 1.1.3 Pruebas de gestión de identidad

Resumen de las pruebas de gestión de la identidad realizadas a la aplicación web de la Institución. La Tabla 10 muestra una lista de resumen de las pruebas realizadas, con el respectivo objetivo de cada una de ellas y las herramientas utilizadas en su respectiva implementación para el análisis.

**Tabla 10.**

*Pruebas de gestión de la identidad*

| CATEGORÍA | CÓDIGO DE PRUEBA | NOMBRE DE LA PRUEBA | OBJETIVO DE LA PRUEBA | HERRAMIENTAS UTILIZADAS |
|---|---|---|---|---|
| Pruebas de gestión de identidad | OTG-IDENT-001 | Definiciones de roles de prueba (OTG-IDENT-001) | El objetivo de la prueba es conocer los diferentes procesos para la creación de roles, las cuales para el caso de la aplicación web, no se crean roles a usuario externos, puesto que, ya se ha creado previamente los mismos. | - |
| | OTG-IDENT-002 | Proceso de registro de usuario de prueba (OTG-IDENT-002) | El objetivo de la prueba es conocer los diferentes procesos de creación de usuarios, ciertas aplicaciones web, realizan este tipo de procesos, de manera automática o semi-automatica, muchas aplicaciones realizan este proceso de forma automática, manejar datos de usuarios de forma manual, se convierte en un proceso bastante complejo de administrar, por lo tanto, no se realiza. | - |
| | OTG-IDENT-003 | Proceso de aprovisionamiento de cuentas de prueba (OTG-IDENT-003) | El objetivo de la prueba es verificar que las cuentas asociadas a la aplicación web son correctas y cuenten | - |

| | | | | |
|---|---|---|---|---|
| | | | la seguridad respectiva para la creación de estas. | |

Fuente: elaboración propia

### 1.1.4 Pruebas de Autenticación

Resumen de las pruebas de autenticación realizadas a la aplicación web de la Institución.

La Tabla 11 muestra una lista de resumen de las pruebas realizadas, con el respectivo objetivo de cada una de ellas y las herramientas utilizadas en su respectiva implementación para el análisis.

**Tabla 11.**

*Pruebas de Autenticación*

| CATEGORÍA | CÓDIGO DE PRUEBA | NOMBRE DE LA PRUEBA | OBJETIVO DE LA PRUEBA | HERRAMIENTAS UTILIZADAS |
|---|---|---|---|---|
| Pruebas de Autenticación | OTG-AUTHN-001 | Prueba de credenciales transportadas a través de un canal cifrado (OTG-AUTHN-001) | El objetivo de la prueba, se la realiza con la finalidad de asegurarse que un ciberdelincuente no consiga obtener información sensible simplemente se rastrea en la red con una herramienta de ataque de vulnerabilidades que alcanza a ser un sniffer. | OWASP Zap |
| | OTG-AUTHN-002 | Prueba de credenciales predeterminadas (OTG-AUTHN-002) | El objetivo de la prueba es verificar que las credenciales hoy en día no se configuran correctamente y las credenciales predeterminadas proporcionadas para la autenticación inicial y configuración nunca son cambiadas, es por ello por lo que no se verifica esta prueba. Credenciales predeterminadas son bien conocidas por los evaluadores de penetración y, es | - |

| | | | | |
|---|---|---|---|---|
| | | | decir, por atacantes maliciosos que logran utilizarlas para obtener acceso a varios tipos de aplicaciones web. | |
| | OTG-AUTHN-003 | Prueba para determinar un mecanismo de bloqueo débil (OTG-AUTHN-003) | El objetivo de la prueba es cubrir todos los aspectos de autenticación donde los mecanismos de bloqueo serían apropiados, por ejemplo, cuando al usuario se le presenten preguntas de seguridad al olvidar su contraseña, al momento de autenticarse. | - |
| | OTG-AUTHN-004 | Prueba para eludir el esquema de autenticación (OTG-AUTHN-004) | El objetivo de la prueba es, tratar que se consiga modificar el parámetro de URL determinado, mediante la manipulación de la forma o por falsificación de las sesiones, se rompa la autenticación, obviando el registro en la página principal y llama directamente a una página externa, que se supone, accederían únicamente después que se realiza la autenticación correctamente. | OWASP Zap |
| | OTG-AUTHN-005 | Prueba de la funcionalidad de recordar contraseña (OTG-AUTHN-005) | El objetivo de la prueba es, tratar que se consiga modificar el parámetro de URL determinado, mediante la manipulación de la forma o por falsificación de las sesiones, se rompa la autenticación, obviando el registro en la página principal y llama directamente a una página | OWASP Zap |

| | | | | |
|---|---|---|---|---|
| | | | externa, que se supone, accederían únicamente después que se realiza la autenticación correctamente. | |
| | OTG-AUTHN-006 | Prueba de la debilidad de la caché del navegador (OTG-AUTHN-006) | El objetivo de la prueba es, verificar que la aplicación web no guarde en caché del navegador usuario y contraseñas, con ello logra un atacante hacer uso de ello e interceptar información dentro de la aplicación web. | OWASP Zap |
| | OTG-AUTHN-007 | Prueba de la política de contraseñas débiles (OTG-AUTHN-007) | El objetivo de la prueba es determinar la resistencia de la aplicación web contra ataques de fuerza bruta y la adivinanza de las contraseñas, para lo cual, se usa diccionarios de contraseñas disponibles en la web y que se llevan a cabo mediante la evaluación de los requerimientos como longitud, complejidad, reutilización y caducidad de las contraseñas. | - |
| | OTG-AUTHN-008 | Prueba de pregunta / respuesta de seguridad débil (OTG-AUTHN-008) | El objetivo de la prueba es analizar si la aplicación web cuenta con la respectiva seguridad en cuanto a las preguntas que se muestran para recuperar una cuenta o a su vez recuperar contraseñas olvidadas. | - |
| | OTG-AUTHN-009 | Prueba de funcionalidades débiles de cambio o restablecimiento | El objetivo de la prueba es verificar si la aplicación permite a los usuarios cambiar o restablecer rápidamente la contraseña sin que | - |

| | | | | |
|---|---|---|---|---|
| | | de contraseña (OTG-AUTHN-009) | un administrador de la aplicación intervenga. | |
| | OTG-AUTHN-010 | Prueba de autenticación más débil en canal alternativo (OTG-AUTHN-010) | El objetivo de la prueba es verificar si la aplicación permite a los usuarios cambiar o restablecer su cuenta y contraseña mediante otros medios, es decir, dispositivos móviles, o a su vez a través de un Centro de llamadas. | - |

Fuente: elaboración propia

### 1.1.5 Pruebas de Autorización

Resumen de las pruebas de autorización realizadas a la aplicación web de la Institución.

La Tabla 12 muestra una lista de resumen de las pruebas realizadas, con el respectivo objetivo de cada una de ellas y las herramientas utilizadas en su respectiva implementación para el análisis.

**Tabla 12.**

*Pruebas de Autorización*

| CATEGORÍA | CÓDIGO DE PRUEBA | NOMBRE DE LA PRUEBA | OBJETIVO DE LA PRUEBA | HERRAMIENTAS UTILIZADAS |
|---|---|---|---|---|
| Pruebas de Autorización | OTG-AUTHZ-001 | Prueba de inclusión de archivos / recorrido de directorio (OTG-AUTHZ-001) | El objetivo de la prueba es conocer si existe o no suficiente seguridad en los diferentes procesos de autorización para usuarios, es decir, el usuario alcanza a acceder a cualquier carpeta desde la raíz de la aplicación web | DotDotpwn |

| | | | | |
|---|---|---|---|---|
| | OTG-AUTHZ-002 | Prueba para eludir el esquema de autorización (OTG-AUTHZ-002) | El objetivo de la prueba es, comprobar cómo se implementó el esquema de autorización para que cada rol o privilegio obtenga acceso a funciones reservadas y recursos de la aplicación web. | - |
| | OTG-AUTHZ-003 | Prueba de escalamiento de privilegios (OTG-AUTHZ-003) | El objetivo de la prueba es identificar si a un atacante mediante esta vulnerabilidad le permite ganar privilegios dentro de la aplicación web, y poder realizar más actividades de las que logra realizar cualquier usuario dentro del sistema. | OWASP Zap |
| | OTG-AUTHZ-004 | Prueba de referencias de objetos directos inseguros (OTG-AUTHZ-004) | El objetivo de la prueba es verificar como un atacante alcanza a modificar de manera fácil los parámetros establecidos por la aplicación. Cuando dicho ataque es exitoso el atacante logra sobrepasar los permisos de autorización y así acceder a recurso dentro del sistema. | OWASP Zap |

Fuente: elaboración propia

### 1.1.6 Pruebas de gestión de sesiones

Resumen de las pruebas de gestión de sesiones realizadas a la aplicación web de la Institución.

La Tabla 13 muestra una lista de resumen de las pruebas realizadas, con el respectivo objetivo de cada una de ellas y las herramientas utilizadas en su respectiva implementación para el análisis.

**Tabla 13.**

*Pruebas de gestión de sesiones*

| CATEGORÍA | CÓDIGO DE PRUEBA | NOMBRE DE LA PRUEBA | OBJETIVO DE LA PRUEBA | HERRAMIENTAS UTILIZADAS |
|---|---|---|---|---|
| Pruebas de gestión de sesiones | OTG-SESS-001 | Prueba para omitir el esquema de administración de sesiones (OTG-SESS-001) | El objetivo de la prueba es conocer la configuración para manejo de sesiones, dentro de la aplicación web, además, se comprobarían que las cookies y otras fichas de sesión se crean de manera segura e impredecible, un atacante es capaz de predecir y falsificar una cookie, para así secuestra la sesión de usuarios que se encuentran legitimante conectados | Burp Suite |
| | OTG-SESS-003 | Prueba de fijación de sesión (OTG-SESS-003) | El objetivo de la prueba es verificar cuando una aplicación no renueva las cookies pertenecientes a la aplicación y respectivamente a la sesión después de una autenticación de usuario exitosa, es posible encontrar una vulnerabilidad de fijación de sesión y de alguna forma obligar al usuario a utilizar una cookie conocida por el atacante. De ser ese el caso, un atacante podría robar la sesión del usuario convirtiéndose ello en un secuestro de sesión. | WebScarab |
| | OTG-SESS-005 | Prueba de falsificación de solicitudes entre sitios | Los tokens de sesión (cookie, ID de sesión, campo oculto), si están expuestos normalmente permitirán a un atacante hacerse pasar por una víctima y acceder a la aplicación de | OWASP Zap |

| | | | | |
|---|---|---|---|---|
| | | (CSRF) (OTG-SESS-005) | forma ilegítima. Es importante que estén protegidos contra escuchas en todo momento, especialmente mientras se encuentran en tránsito entre el navegador del cliente y los servidores de aplicaciones. | |
| | OTG-SESS-006 | Prueba de la funcionalidad de cierre de sesión (OTG-SESS-006) | La terminación de la sesión es una parte importante del ciclo de vida de la sesión. Reducir al mínimo la vida útil de los tokens de sesión disminuye la probabilidad de un ataque de secuestro de sesión exitoso. Esto alcanza a verse como un control para evitar otros ataques como Cross Site Scripting y Cross Site Request Forgery. Se sabe que tales ataques dependen de que un usuario tenga una sesión autenticada presente. No tener una terminación de sesión segura solo aumenta la superficie de ataque para cualquiera de estos ataques. | Burp Suite |

Fuente: elaboración propia

### 1.1.7 Pruebas de validación de entradas

Resumen de las pruebas de validación de entrada realizadas a la aplicación web de la Institución. La Tabla 14 muestra una lista de resumen de las pruebas realizadas, con el respectivo objetivo de cada una de ellas y las herramientas utilizadas en su respectiva implementación para el análisis.

**Tabla 14.**

*Pruebas de validación de entradas*

| CATEGORÍA | CÓDIGO DE PRUEBA | NOMBRE DE LA PRUEBA | OBJETIVO DE LA PRUEBA | HERRAMIENTAS UTILIZADAS |
|---|---|---|---|---|
| Pruebas de validación de entrada | OTG-INPVAL-001 | Prueba de secuencias de comandos de sitios cruzados reflejados (OTG-INPVAL-001) | El objetivo de la prueba es verificar la correcta codificación de los caracteres, en muchos de los casos los servidores pueden filtrar algunas codificaciones de caracteres, esta vulnerabilidad mediante XSS, la cual, un atacante utiliza una aplicación web para de esa manera enviar algún código malicioso, generalmente en forma de un script del lado del navegador, a un usuario final diferente. | Burp Suite |
| | OTG-INPVAL-003 | Prueba de manipulación verbos (OTG-INPVAL-003) | El objetivo de la prueba determinar posee métodos alternativos a GET Y POST para el envío y transmisión de información, estos métodos alternativos de igual forma responden a solicitudes desde la aplicación web de una manera no prevista por los desarrolladores, cuando estos son puestos en marcha. | Curl |

| | | | | |
|---|---|---|---|---|
| | OTG-INPVAL-004 | Prueba de contaminación de parámetros HTTP (OTG-INPVAL-004) | El objetivo de la prueba determinar si existe alguna manera de realizar un ataque de contaminación de parámetros que afecten al comportamiento de los formularios del lado del cliente, es decir, por medio de parámetros quienes se encargan de enviar dichos datos a la aplicación web mediante urls, para que esta funcione aparentemente de manera adecuada, generalmente, se ve en formularios de ingreso de usuarios, para enviar correos electrónicos, para comentar, llenar encuestas, realizar búsquedas de información entre otros. | OWASP Zap |
| | OTG-INPVAL-005 | Prueba de inyección SQL (OTG-INPVAL-005) | El objetivo de la prueba es verificar si existe la manera de realizar el ataque de inyección SQL, mediante este ataque a la aplicación web, se alcanza a leer datos confidenciales de una base de datos, además, modificar los datos de la base de datos mediante los procesos de (insertar / actualizar / eliminar), ejecutar operaciones administrativas en la base de datos como, por ejemplo, cerrar el DBMS (Database Management System), entre otras operaciones sobre la base de datos. | SqlMap |

| | | | | |
|---|---|---|---|---|
| | OTG-INPVAL-013 | Prueba de inyección de comando (OTG-INPVAL-013) | El objetivo de la prueba es comprobar una aplicación para la inyección de comandos del sistema operativo, esta prueba intenta inyectar un comando del sistema operativo web a través de una solicitud HTTP a la aplicación. | - |

Fuente: elaboración propia

### 1.1.8 Pruebas de manejos de errores

Resumen de las pruebas de manejos de errores realizadas a la aplicación web de la Institución.
La Tabla 15 muestra una lista de resumen de las pruebas realizadas, con el respectivo objetivo de cada una de ellas y las herramientas utilizadas en su respectiva implementación para el análisis.

**Tabla 15.**

*Pruebas de manejo de errores*

| CATEGORÍA | CÓDIGO DE PRUEBA | NOMBRE DE LA PRUEBA | OBJETIVO DE LA PRUEBA | HERRAMIENTAS UTILIZADAS |
|---|---|---|---|---|
| Pruebas de manejo de errores | OTG-ERR-001 | Análisis de códigos de error (OTG-ERR-001) | El objetivo de la prueba es determinar si una página de la aplicación web, contiene algún control a prueba de errores, ya sean estas por conexión o parámetros enviados de manera incorrecta, o algún problema en cuanto a respuesta desde el servidor web. | Burp Suite, Google |

Fuente: elaboración propia

### 1.1.9 Pruebas de criptografía débil

Resumen de las pruebas de manejos de criptografía débil realizadas a la aplicación web de la Institución.
La Tabla 16 muestra una lista de resumen de las pruebas realizadas, con el respectivo objetivo de cada una de ellas y las herramientas utilizadas en su respectiva implementación para el análisis.

**Tabla 16.**

*Pruebas de criptografía débil*

| CATEGORÍA | CÓDIGO DE PRUEBA | NOMBRE DE LA PRUEBA | OBJETIVO DE LA PRUEBA | HERRAMIENTAS UTILIZADAS |
|---|---|---|---|---|
| Pruebas de criptografía débil | OTG-CRYPST-001 | Prueba de cifrados SSL / TLS débiles, protección insuficiente de la capa de transporte (OTG-CRYPST-001) | El objetivo de la prueba es determinar si la aplicación web, contiene un certificado TLS/SSL (Transport Layer Security) / (Secure Sockets Layer), la cual, ayuda a mantener segura la aplicación web, además, se verifica si la información que se envía entre el usuario y la aplicación web, para que estas viajen de manera segura. | Nmap, testssl.sh |
| | OTG-CRYPST-003 | Prueba de información confidencial enviada a través de canales no cifrados (OTG-CRYPST-003) | El objetivo de la prueba es, determinar si los datos que se envía a través de la red son cifrados, si los datos que se envían, se los realiza mediante HTTPS (Hypertext Transfer Protocol Secure), este mecanismo de protección no tendría limitaciones o vulnerabilidades. | testssl.sh |

Fuente: elaboración propia

## 1.2 Evaluación de Riesgos

La metodología OWASP indica que, en cada prueba se tiene una evaluación de riego según su criticidad, la cual, se detallara a continuación, se toma en cuenta varios factores para evaluar dicha característica en cada prueba realizada.

Para realizar este cálculo progresivo, se toma en cuenta los siguientes factores, los cuales posteriormente, se tomarán en cuenta para el cálculo del riesgo en cada una de las pruebas a realizarse:

- **Factores de Probabilidad**
    - Habilidades Requeridas: Indica si se requiere alguna habilidad para descubrir una posible vulnerabilidad y sus valores son: Sin habilidades técnicas (1), algunas habilidades técnicas (3), usuario avanzado de computadoras (5), habilidades de programación y redes (6), habilidades de penetración de seguridad (9).
    - Motivo: Si es de gran importancia la información encontrada y sus valores son: Recompensa baja o nula (1), posible recompensa (4), recompensa alta (9).
    - Oportunidad: Indica si se requiere o no algún tipo de permiso para lograr acceder y buscar información sobre una posible vulnerabilidad existente y sus valores son: Se requiere acceso total o recursos costosos (0), se requiere acceso o recursos especiales (4), se requiere algún acceso o recursos (7), no se requiere acceso o recursos (9).
    - Tamaño de la Población: Usuarios que se consiguen verse afectados ante una posible amenaza y sus valores son Desarrolladores (2), administradores de sistemas (2), usuarios de intranet (4), socios (5), usuarios autenticados (6), usuarios de Internet anónimos (9).
- **Factores de Impacto técnico**
    - Pérdida de Confidencialidad: Si el impacto de una posible vulnerabilidad determinara una posible pérdida de la confidencialidad de la información, hacia el usuario y sus valores son: Información mínima no confidencial divulgada (2), información mínima crítica divulgada (6), divulgación de datos no confidenciales extensos (6), divulgación de datos críticos extensos (7), divulgación de todos los datos (9).
    - Pérdida de Integridad: Se refiere a la pérdida o alteración de la información a causa de una posible vulnerabilidad existente y sus valores son: Datos mínimos levemente corruptos (1), mínimos datos muy corruptos (3), datos extensos ligeramente corruptos (5), datos extensos muy corruptos (7), todos los datos totalmente corruptos (9).

- Pérdida de Disponibilidad: Indica si la información se podrá perder y no tener acceso a la misma ante una posible vulnerabilidad conocida y sus valores son Servicios secundarios mínimos interrumpidos (1), servicios primarios mínimos interrumpidos (5), servicios secundarios extensos interrumpidos (5), servicios primarios extensos interrumpidos (7), todos los servicios completamente perdidos (9).
- Pérdida de Responsabilidad: Ante un posible ataque por un ciberdelincuente su objetivo realizado será o no confidencial y sus valores son: Totalmente rastreable (1), posiblemente rastreable (7), completamente anónimo (9).

- **Factores de vulnerabilidad**
  - Fácil de Descubrir: El nivel de dificultad para descubrir una vulnerabilidad y sus valores son: Prácticamente imposible (1), difícil (3), fácil (7), herramientas automatizadas disponibles (9).
  - Fácil Explotación: El nivel de dificultad para explotar una vulnerabilidad y sus valores son: Teórico (1), difícil (3), fácil (5), herramientas automatizadas disponibles (9).
  - Conciencia: Indica si la vulnerabilidad encontrada es visual o no y sus valores son: Desconocido (1), oculto (4), obvio (6), conocimiento público (9).
  - Detección de Intrusiones: Indica la posible vulnerabilidad existente detecta o no un ataque y sus valores son: Detección activa en la aplicación (1), registrada y revisada (3), registrada sin revisión (8), no registrada (9).
- **Factores de impacto empresarial**
  - Daño financiero: Refiere al factor económico que podría ocasionar un posible ataque exitoso a la aplicación web y sus valores son: Menos que el costo de reparar la vulnerabilidad (1), efecto menor en la ganancia anual (3), efecto significativo en la ganancia anual (7), quiebra (9).
  - Daño a la reputación: Indica si un posible ataque a la aplicación web, logra ocasionar el desprestigio de usuarios o inclusive a la institución y sus valores son: Daño mínimo (1), pérdida de cuentas importantes (4), pérdida de buena voluntad (5), daño a la marca (9).
  - Incumplimiento: Contempla si existiera una posible infracción ante una amenaza que se encuentre dentro de la aplicación web y sus valores son: Violación menor (2), violación clara (5), violación de alto perfil (7).

- Violación de la privacidad: Se refiere a la cantidad de usuarios que se verían afectados ante un posible ataque a la aplicación web y sus valores son: Un individuo (3), cientos de personas (5), miles de personas (7), millones de personas (9).

Es fundamental tomar en cuenta estos factores para determinar la severidad de una vulnerabilidad, de ello dependerá la correcta determinación de un problema a ser solucionado, con la finalidad de tener un sistema libre de ataques, los cuales alcanzan a ser expuestos por una aplicación web que no tenga una correcta seguridad en cuanto al manejo de la información se refiere.

Estos factores se evalúan y se obtiene de acuerdo con el resultado de severidad de riesgo de una determinada vulnerabilidad, los criterios de evaluación se toman en cuenta de acuerdo con los análisis de cada prueba realizada y los valores de calificación son valores estándares indicados por la metodología de calificación de riesgo de OWASP.

A continuación, en la Tabla 17, se muestra la tabla de la estimación de probabilidad y la estimación del impacto se combinan para calcular la gravedad general de este riesgo.

**Tabla 17.**

*Niveles de la gravedad del riesgo*

| Likelihood and Impact Levels | |
|---|---|
| 0 to <3 | LOW |
| 3 to <6 | MEDIUM |
| 6 to 9 | HIGH |

OWASP. Recuperado 03/12/2020. Risk Rating Methodology. https://owasp.org/www-community/OWASP_Risk_Rating_Methodology

Es necesario poder estimar la severidad de los riesgos hacia la Institución por parte de una determinada aplicación web, y hacer una decisión informada sobre aquellos riesgos que, se encuentren presentes en una aplicación web.

A continuación, en la Tabla 18, se muestra la gravedad de los riesgos, que se toma en cuenta en la probabilidad e impacto.

**Tabla 18.**

*Determinación de la gravedad del riesgo*

| Overall Risk Severity | | | | |
|---|---|---|---|---|
| Impact | HIGH | Medium | High | Critical |
| | MEDIUM | Low | Medium | High |
| | LOW | Note | Low | Medium |
| | | LOW | MEDIUM | HIGH |
| | Likelihood | | | |

OWASP. Recuperado 03/12/2020. Risk Rating Methodology. https://owasp.org/www-community/OWASP_Risk_Rating_Methodology

En esta sección, se muestra los resultados de la evaluación de las diferentes pruebas que sugiere la metodología OWASP.

### 3.2.1 Recopilación de la Información

A continuación, en la Tabla 19, se detalla el valor de la evaluación de riego de la prueba realizada, en la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 19.**

*Realizar el descubrimiento y reconocimiento del motor de búsqueda para la fuga de información (OTG-INFO-001)*

**Evaluación de riesgos de OWASP**
**Realizar el descubrimiento y reconocimiento del motor de búsqueda para la fuga de información (OTG-INFO-001)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Sin conocimientos técnicos [1] | 1 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | No aplica [0] | 0 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| | | | | | |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Fácil [7] | 7 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Herramientas automatizadas disponibles [9] | 9 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Conocimiento público [9] | 9 | Incumplimiento | Violación de alto perfil [7] | 7 |
| Detección de intrusiones | Detección activa en la aplicación [1] | 1 | Violación de privacidad | Un individuo [3] | 3 |

| Puntuación de probabilidad: | 5 | Puntuación de impacto: | 2.875 |
|---|---|---|---|

Severidad general del riesgo: **Bajo**

| Probabilidad | Impacto ->Bajo<- | Medio | Alto |
|---|---|---|---|
| Bajo | Ninguno | Bajo | Medio |
| ->Medio<- | ->Bajo<- | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 20, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor medio en la severidad de riesgo.

**Tabla 20.**

*Servidor web de huellas digitales (OTG-INFO-002)*

**Evaluación de riesgos de OWASP**
**Servidor web de huellas digitales (OTG-INFO-002)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Habilidades de penetración de la seguridad [9] | 9 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Gran recompensa [9] | 9 | Pérdida de integridad | Datos extensos ligeramente corruptos [5] | 5 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Fácil [7] | 7 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Fácil [5] | 5 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | Violación de alto perfil [7] | 7 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | Un individuo [3] | 3 |
| **Puntuación de probabilidad:** | **5.125** | | **Puntuación de impacto:** | **3.875** | |

**Severidad general del riesgo:** **Medio**

| Probabilidad \ Impacto | Bajo | ->Medio<- | Alto |
|---|---|---|---|
| Bajo | Ninguno | **Bajo** | Medio |
| ->Medio<- | **Bajo** | ->Medio<- | Alto |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 21, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 21.**

*Revisar metarchivos del servidor web para detectar fugas de información (OTG-INFO-003)*

**Evaluación de riesgos de OWASP**
**Revisar metarchivos del servidor web para detectar fugas de información (OTG-INFO-003)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | No aplica [0] | 0 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | No se requiere acceso ni recursos [9] | 9 | Pérdida de disponibilidad | Servicios secundarios mínimos interrumpidos [1] | 1 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Fácil [7] | 7 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Fácil [5] | 5 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | Violación de alto perfil [7] | 7 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | Un individuo [3] | 3 |
| **Puntuación de probabilidad:** | **3.125** | | **Puntuación de impacto:** | **2.875** | |

**Severidad general del riesgo:** **Bajo**

| Probabilidad \ Impacto | ->Bajo<- | Medio | Alto |
|---|---|---|---|
| Bajo | Ninguno | **Bajo** | Medio |
| ->Medio<- | **->Bajo<-** | Medio | Alto |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 22, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor medio en la severidad de riesgo.

**Tabla 22.**

*Enumerar aplicaciones en el servidor web (OTG-INFO-004)*

**Evaluación de riesgos de OWASP**
**Enumerar aplicaciones en el servidor web (OTG-INFO-004)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Sin conocimientos técnicos [1] | 1 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Servicios secundarios mínimos interrumpidos [1] | 1 |
| Tamaño de la poblacion | Usuarios de Internet anónimos [9] | 9 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Fácil [7] | 7 | Daño financiero | Efecto significativo en el beneficio anual [7] | 7 |
| Facilidad de explotación | Fácil [5] | 5 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Obvio [6] | 6 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | Un individuo [3] | 3 |

| **Puntuación de probabilidad:** | **4.875** | **Puntuación de impacto:** | **3** |
|---|---|---|---|

**Severidad general del riesgo:** **Medio**

| **Probabilidad** | **Impacto** Bajo | **->Medio<-** | Alto |
|---|---|---|---|
| Bajo | Ninguno | **Bajo** | Medio |
| **->Medio<-** | **Bajo** | **->Medio<-** | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 23, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 23.**

*Revisión de los comentarios y metadatos de la página web para detectar fugas de información (OTG-INFO-005)*

**Evaluación de riesgos de OWASP**
**Revisión de los comentarios y metadatos de la página web para detectar fugas de información (OTG-INFO-005)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Sin conocimientos técnicos [1] | 1 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | No aplica [0] | 0 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Teórico [1] | 1 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | No aplica [0] | 0 |

| **Puntuación de probabilidad:** | **0.875** | **Puntuación de impacto:** | **1.625** |
|---|---|---|---|

**Severidad general del riesgo:** Ninguno

| **Probabilidad** | **Impacto** **->Bajo<-** | Medio | Alto |
|---|---|---|---|
| **->Bajo<-** | **->Ninguno<-** | **Bajo** | Medio |
| Medio | **Bajo** | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 24, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor alto en la severidad de riesgo.

**Tabla 24.**

*Identificar los puntos de entrada de la aplicación (OTG-INFO-006)*

**Evaluación de riesgos de OWASP**
**Identificar los puntos de entrada de la aplicación (OTG-INFO-006)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Algunas habilidades técnicas [3] | 3 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Gran recompensa [9] | 9 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Se interrumpieron amplios servicios primarios [7] | 7 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Herramientas automatizadas disponibles [9] | 9 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Fácil [5] | 5 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | Registrado sin revisión [8] | 8 | Violación de privacidad | Un individuo [3] | 3 |

| **Puntuación de probabilidad:** | **6.375** | **Puntuación de impacto:** | **4** |
|---|---|---|---|

**Severidad general del riesgo:** **Alto**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | Bajo | **->Medio<-** | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| Medio | **Bajo** | Medio | Alto |
| **->Alto<-** | Medio | **->Alto<-** | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 25, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor medio en la severidad de riesgo.

**Tabla 25.**

*Mapa de rutas de ejecución a través de la aplicación (OTG-INFO-007)*

**Evaluación de riesgos de OWASP**
**Mapa de rutas de ejecución a través de la aplicación (OTG-INFO-007)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Algunas habilidades técnicas [3] | 3 | Pérdida de confidencialidad | No aplica [0] | 0 |
| Motivo | Gran recompensa [9] | 9 | Pérdida de integridad | Datos extensos ligeramente corruptos [5] | 5 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Se interrumpieron amplios servicios primarios [7] | 7 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Fácil [7] | 7 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Clara infracción [5] | 5 |
| Detección de intrusiones | Detección activa en la aplicación [1] | 1 | Violación de privacidad | Cientos de personas [5] | 5 |

| **Puntuación de probabilidad:** | **5** | **Puntuación de impacto:** | **3.125** |
|---|---|---|---|

**Severidad general del riesgo:** **Medio**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | Bajo | **->Medio<-** | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| **->Medio<-** | **Bajo** | **->Medio<-** | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 26, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 26.**

*Marco de aplicación web de huellas dactilares (OTG-INFO-008)*

**Evaluación de riesgos de OWASP**
**Marco de aplicación web de huellas dactilares (OTG-INFO-008)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Algunas habilidades técnicas [3] | 3 | Pérdida de confidencialidad | No aplica [0] | 0 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Servicios secundarios mínimos interrumpidos [1] | 1 |
| Tamaño de la poblacion | Usuarios de Internet anónimos [9] | 9 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Fácil [7] | 7 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Clara infracción [5] | 5 |
| Detección de intrusiones | Detección activa en la aplicación [1] | 1 | Violación de privacidad | Un individuo [3] | 3 |

| **Puntuación de probabilidad:** | **4.75** | **Puntuación de impacto:** | **2.375** |
|---|---|---|---|

**Severidad general del riesgo:** **Bajo**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | **->Bajo<-** | Medio | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| **->Medio<-** | **->Bajo<-** | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 27, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor medio en la severidad de riesgo.

**Tabla 27.**

*Aplicación web de huellas dactilares (OTG-INFO-009)*

**Evaluación de riesgos de OWASP**
**Aplicación web de huellas dactilares (OTG-INFO-009)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Se divulgan numerosos datos críticos [7] | 7 |
| Motivo | Gran recompensa [9] | 9 | Pérdida de integridad | Datos extensos ligeramente corruptos [5] | 5 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios de intranet [4] | 4 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Clara infracción [5] | 5 |
| Detección de intrusiones | Detección activa en la aplicación [1] | 1 | Violación de privacidad | Un individuo [3] | 3 |

| **Puntuación de probabilidad:** | **4.5** | **Puntuación de impacto:** | **3.5** |
|---|---|---|---|

**Severidad general del riesgo:** **Medio**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | Bajo | **->Medio<-** | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| **->Medio<-** | **Bajo** | **->Medio<-** | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

### 3.2.2 Pruebas de gestión de la configuración y la implementación

A continuación, en la Tabla 28, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 28.**

*Prueba de configuración de red / infraestructura (OTG-CONFIG-001)*

**Evaluación de riesgos de OWASP**
**Prueba de configuración de red / infraestructura (OTG-CONFIG-001)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | No aplica [0] | 0 | Pérdida de confidencialidad | No aplica [0] | 0 |
| Motivo | No aplica [0] | 0 | Pérdida de integridad | No aplica [0] | 0 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | No aplica [0] | 0 |
| Tamaño de la poblacion | No aplica [0] | 0 | Pérdida de responsabilidad | No aplica [0] | 0 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | No aplica [0] | 0 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | No aplica [0] | 0 | Daño a la reputación | No aplica [0] | 0 |
| Conciencia | No aplica [0] | 0 | Incumplimiento | No aplica [0] | 0 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | No aplica [0] | 0 |
| **Puntuación de probabilidad:** | **0** | | **Puntuación de impacto:** | **0** | |

**Severidad general del riesgo:** Ninguno

| Probabilidad | Impacto ->Bajo<- | Medio | Alto |
|---|---|---|---|
| ->Bajo<- | ->Ninguno<- | Bajo | Medio |
| Medio | Bajo | Medio | Alto |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 29, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 29.**

*Configuración de la plataforma de aplicaciones de prueba (OTG-CONFIG-002)*

**Evaluación de riesgos de OWASP**
**Configuración de la plataforma de aplicaciones de prueba (OTG-CONFIG-002)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | No aplica [0] | 0 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | Teórico [1] | 1 | Daño a la reputación | No aplica [0] | 0 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | No aplica [0] | 0 |
| Detección de intrusiones | Detección activa en la aplicación [1] | 1 | Violación de privacidad | No aplica [0] | 0 |
| **Puntuación de probabilidad:** | **3.5** | | **Puntuación de impacto:** | **1.25** | |

**Severidad general del riesgo:** Bajo

| Probabilidad | Impacto ->Bajo<- | Medio | Alto |
|---|---|---|---|
| Bajo | Ninguno | Bajo | Medio |
| ->Medio<- | ->Bajo<- | Medio | Alto |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 30, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 30.**

*Manejo de extensiones de archivo de prueba para información confidencial (OTG-CONFIG-003)*

**Evaluación de riesgos de OWASP**
**Manejo de extensiones de archivo de prueba para información confidencial (OTG-CONFIG-003)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Algunas habilidades técnicas [3] | 3 | Pérdida de confidencialidad | No aplica [0] | 0 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | No aplica [0] | 0 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | No aplica [0] | 0 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | No aplica [0] | 0 |
| | | | | | |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | Teórico [1] | 1 | Daño a la reputación | No aplica [0] | 0 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | No aplica [0] | 0 |
| Detección de intrusiones | Detección activa en la aplicación [1] | 1 | Violación de privacidad | No aplica [0] | 0 |

| **Puntuación de probabilidad:** | **2.375** | **Puntuación de impacto:** | **0** |
|---|---|---|---|

**Severidad general del riesgo:** Ninguno

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | **->Bajo<-** | Medio | Alto |
| **->Bajo<-** | **->Ninguno<-** | **Bajo** | Medio |
| Medio | **Bajo** | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 31, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 31.**

*Revisar archivos antiguos, de copia de seguridad y sin referencia en busca de información confidencial (OTG-CONFIG-004)*

**Evaluación de riesgos de OWASP**
**Revisar archivos antiguos, de copia de seguridad y sin referencia en busca de información confidencial (OTG-CONFIG-004)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | No aplica [0] | 0 | Pérdida de confidencialidad | No aplica [0] | 0 |
| Motivo | No aplica [0] | 0 | Pérdida de integridad | No aplica [0] | 0 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | No aplica [0] | 0 |
| Tamaño de la poblacion | No aplica [0] | 0 | Pérdida de responsabilidad | No aplica [0] | 0 |
| | | | | | |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | No aplica [0] | 0 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | No aplica [0] | 0 | Daño a la reputación | No aplica [0] | 0 |
| Conciencia | No aplica [0] | 0 | Incumplimiento | No aplica [0] | 0 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | No aplica [0] | 0 |

| **Puntuación de probabilidad:** | **0** | **Puntuación de impacto:** | **0** |
|---|---|---|---|

**Severidad general del riesgo:** Ninguno

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | **->Bajo<-** | Medio | Alto |
| **->Bajo<-** | **->Ninguno<-** | **Bajo** | Medio |
| Medio | **Bajo** | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 32, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 32.**

*Infraestructura de enumeración e interfaces de Administración de Aplicaciones (OTG-CONFIG-005)*

**Evaluación de riesgos de OWASP**
**Enumerar las interfaces de administración de aplicaciones e infraestructura (OTG-CONFIG-005)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Algunas habilidades técnicas [3] | 3 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | Servicios secundarios mínimos interrumpidos [1] | 1 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Herramientas automatizadas disponibles [9] | 9 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | Detección activa en la aplicación [1] | 1 | Violación de privacidad | Cientos de personas [5] | 5 |

| Puntuación de probabilidad: | 3.875 | Puntuación de impacto: | 2.75 |
|---|---|---|---|

Severidad general del riesgo: **Bajo**

| Probabilidad \ Impacto | ->Bajo<- | Medio | Alto |
|---|---|---|---|
| Bajo | Ninguno | **Bajo** | Medio |
| ->Medio<- | ->Bajo<- | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 33, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor medio en la severidad de riesgo.

**Tabla 33.**

*Prueba de Métodos HTTP (OTG-CONFIG-006)*

**Evaluación de riesgos de OWASP**
**Probar métodos HTTP (OTG-CONFIG-006)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Se divulgan numerosos datos críticos [7] | 7 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Numerosos datos muy corruptos [7] | 7 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Se interrumpieron amplios servicios primarios [7] | 7 |
| Tamaño de la poblacion | Usuarios de intranet [4] | 4 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Fácil [7] | 7 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Fácil [5] | 5 | Daño a la reputación | Pérdida de cuentas importantes [4] | 4 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | Detección activa en la aplicación [1] | 1 | Violación de privacidad | Cientos de personas [5] | 5 |

| Puntuación de probabilidad: | 4.625 | Puntuación de impacto: | 5.25 |
|---|---|---|---|

Severidad general del riesgo: **Medio**

| Probabilidad \ Impacto | Bajo | ->Medio<- | Alto |
|---|---|---|---|
| Bajo | Ninguno | **Bajo** | Medio |
| ->Medio<- | **Bajo** | ->Medio<- | Alto |
| Alto | Medio | Alto | Crítico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 34, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor alto en la severidad de riesgo.

**Tabla 34.**

*Prueba de Seguridad de Transporte Estricto HTTP - HSTS (OTG-CONFIG-007)*

**Evaluación de riesgos de OWASP**
**Probar la seguridad de transporte estricta de HTTP (OTG-CONFIG-007)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | No se requiere acceso ni recursos [9] | 9 | Pérdida de disponibilidad | Se interrumpieron amplios servicios primarios [7] | 7 |
| Tamaño de la poblacion | No aplica [0] | 0 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Herramientas automatizadas disponibles [9] | 9 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Herramientas automatizadas disponibles [9] | 9 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Obvio [6] | 6 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | Cientos de personas [5] | 5 |

| **Puntuación de probabilidad:** | **6.375** | **Puntuación de impacto:** | **4.25** |
|---|---|---|---|

**Severidad general del riesgo:** **Alto**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | Bajo | **->Medio<-** | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| Medio | **Bajo** | Medio | Alto |
| **->Alto<-** | Medio | **->Alto<-** | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 35, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor alto en la severidad de riesgo.

**Tabla 35.**

*Prueba de Política de Dominio cruzado RIA (OTG-CONFIG-008)*

**Evaluación de riesgos de OWASP**
**Probar la política de dominio cruzado de RIA (OTG-CONFIG-008)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Todos los datos divulgados [9] | 9 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Todos los datos están totalmente corruptos [9] | 9 |
| Oportunidad | No se requiere acceso ni recursos [9] | 9 | Pérdida de disponibilidad | Todos los servicios completamente perdidos [9] | 9 |
| Tamaño de la poblacion | Usuarios de Internet anónimos [9] | 9 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Herramientas automatizadas disponibles [9] | 9 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Herramientas automatizadas disponibles [9] | 9 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Obvio [6] | 6 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | Cientos de personas [5] | 5 |

| **Puntuación de probabilidad:** | **7.5** | **Puntuación de impacto:** | **5.625** |
|---|---|---|---|

**Severidad general del riesgo:** **Alto**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | Bajo | **->Medio<-** | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| Medio | **Bajo** | Medio | Alto |
| **->Alto<-** | Medio | **->Alto<-** | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

### 3.2.3 Pruebas de gestión de la identidad

A continuación, en la Tabla 36, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 36.**

*Definiciones de roles de prueba (OTG-IDENT-001)*

**Evaluación de riesgos de OWASP**
**Definiciones de roles de prueba (OTG-IDENT-001)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | No aplica [0] | 0 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | No aplica [0] | 0 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | No aplica [0] | 0 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | No aplica [0] | 0 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | No aplica [0] | 0 |

| **Puntuación de probabilidad:** | **2.125** | **Puntuación de impacto:** | **1** |
|---|---|---|---|

**Severidad general del riesgo:** Ninguno

| | **Impacto** | | | |
|---|---|---|---|---|
| **Probabilidad** | **->Bajo<-** | Medio | Alto | |
| **->Bajo<-** | **->Ninguno<-** | **Bajo** | Medio | |
| Medio | **Bajo** | Medio | Alto | |
| Alto | Medio | Alto | Critico | |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 37, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 37.**

*Proceso de registro de usuario de prueba (OTG-IDENT-002)*

**Evaluación de riesgos de OWASP**
**Proceso de registro de usuario de prueba (OTG-IDENT-002)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | No aplica [0] | 0 | Pérdida de confidencialidad | No aplica [0] | 0 |
| Motivo | No aplica [0] | 0 | Pérdida de integridad | No aplica [0] | 0 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | No aplica [0] | 0 |
| Tamaño de la poblacion | No aplica [0] | 0 | Pérdida de responsabilidad | No aplica [0] | 0 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | No aplica [0] | 0 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | No aplica [0] | 0 | Daño a la reputación | No aplica [0] | 0 |
| Conciencia | No aplica [0] | 0 | Incumplimiento | No aplica [0] | 0 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | No aplica [0] | 0 |

| **Puntuación de probabilidad:** | **0** | **Puntuación de impacto:** | **0** |
|---|---|---|---|

**Severidad general del riesgo:** Ninguno

| | **Impacto** | | | |
|---|---|---|---|---|
| **Probabilidad** | **->Bajo<-** | Medio | Alto | |
| **->Bajo<-** | **->Ninguno<-** | **Bajo** | Medio | |
| Medio | **Bajo** | Medio | Alto | |
| Alto | Medio | Alto | Critico | |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 38, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 38.**

*Proceso de aprovisionamiento de cuentas de prueba (OTG-IDENT-003)*

**Evaluación de riesgos de OWASP**
**Proceso de aprovisionamiento de cuentas de prueba (OTG-IDENT-003)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Sin conocimientos técnicos [1] | 1 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | No aplica [0] | 0 |
| Conciencia | Conocimiento público [9] | 9 | Incumplimiento | No aplica [0] | 0 |
| Detección de intrusiones | Registrado y revisado [3] | 3 | Violación de privacidad | No aplica [0] | 0 |
| **Puntuación de probabilidad:** | **3.25** | | **Puntuación de impacto:** | **1.125** | |

**Severidad general del riesgo:** **Bajo**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | **->Bajo<-** | Medio | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| **->Medio<-** | **->Bajo<-** | Medio | Alto |
| Alto | Medio | Alto | Critico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

### 3.2.4 Pruebas de Autenticación

A continuación, en la Tabla 39, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor alto en la severidad de riesgo.

**Tabla 39.**

*Prueba de credenciales transportadas a través de un canal cifrado (OTG-AUTHN-001)*

**Evaluación de riesgos de OWASP**
**Prueba de credenciales transportadas a través de un canal cifrado (OTG-AUTHN-001)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Todos los datos divulgados [9] | 9 |
| Motivo | Gran recompensa [9] | 9 | Pérdida de integridad | Datos extensos ligeramente corruptos [5] | 5 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Se interrumpieron amplios servicios primarios [7] | 7 |
| Tamaño de la poblacion | Usuarios de Internet anónimos [9] | 9 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | Efecto significativo en el beneficio anual [7] | 7 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de buena voluntad [5] | 5 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Clara infracción [5] | 5 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | Cientos de personas [5] | 5 |
| **Puntuación de probabilidad:** | **5** | | **Puntuación de impacto:** | **6.25** | |

**Severidad general del riesgo:** **Alto**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | Bajo | Medio | **->High<-** |
| Bajo | Ninguno | **Bajo** | Medio |
| **->Medio<-** | **Bajo** | Medio | **->Alto<-** |
| Alto | Medio | Alto | Critico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 40, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor medio en la severidad de riesgo.

**Tabla 40.**

*Prueba de credenciales predeterminadas (OTG-AUTHN-002)*

**Evaluación de riesgos de OWASP**
**Prueba de credenciales predeterminadas (OTG-AUTHN-002)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Habilidades de penetración de la seguridad [9] | 9 | Pérdida de confidencialidad | Todos los datos divulgados [9] | 9 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos extensos ligeramente corruptos [5] | 5 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | Se interrumpieron amplios servicios primarios [7] | 7 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | No aplica [0] | 0 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | No aplica [0] | 0 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | Cientos de personas [5] | 5 |

| Puntuación de probabilidad: | 3.875 | Puntuación de impacto: | 4.125 |
|---|---|---|---|

Severidad general del riesgo: **Medio**

| Probabilidad \ Impacto | Bajo | ->Medio<- | Alto |
|---|---|---|---|
| Bajo | Ninguno | Bajo | Medio |
| ->Medio<- | Bajo | ->Medio<- | Alto |
| Alto | Medio | Alto | Crítico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 41, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 41.**

*Prueba para determinar un mecanismo de bloqueo débil (OTG-AUTHN-003)*

**Evaluación de riesgos de OWASP**
**Prueba para determinar un mecanismo de bloqueo débil (OTG-AUTHN-003)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | No aplica [0] | 0 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios de Internet anónimos [9] | 9 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | No aplica [0] | 0 |
| Conciencia | No aplica [0] | 0 | Incumplimiento | No aplica [0] | 0 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | Cientos de personas [5] | 5 |

| Puntuación de probabilidad: | 2 | Puntuación de impacto: | 1.75 |
|---|---|---|---|

Severidad general del riesgo: **Ninguno**

| Probabilidad \ Impacto | ->Bajo<- | Medio | Alto |
|---|---|---|---|
| ->Bajo<- | ->Ninguno<- | Bajo | Medio |
| Medio | Bajo | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 42, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor medio en la severidad de riesgo.

**Tabla 42.**

*Prueba para eludir el esquema de autenticación (OTG-AUTHN-004)*

**Evaluación de riesgos de OWASP**
**Prueba para eludir el esquema de autenticación (OTG-AUTHN-004)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Sin conocimientos técnicos [1] | 1 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | Efecto significativo en el beneficio anual [7] | 7 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de cuentas importantes [4] | 4 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | Cientos de personas [5] | 5 |

| Puntuación de probabilidad: | 4.625 | Puntuación de impacto: | 3.625 |
|---|---|---|---|

Severidad general del riesgo: **Medio**

| Probabilidad \ Impacto | Bajo | ->Medio<- | Alto |
|---|---|---|---|
| Bajo | Ninguno | Bajo | Medio |
| ->Medio<- | Bajo | ->Medio<- | Alto |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 43, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor alto en la severidad de riesgo.

**Tabla 43.**

*Prueba de la funcionalidad de recordar contraseña (OTG-AUTHN-005)*

**Evaluación de riesgos de OWASP**
**Prueba de la funcionalidad de recordar contraseña (OTG-AUTHN-005)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Se divulgan numerosos datos críticos [7] | 7 |
| Motivo | Gran recompensa [9] | 9 | Pérdida de integridad | Numerosos datos muy corruptos [7] | 7 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | Se interrumpieron amplios servicios primarios [7] | 7 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | Efecto significativo en el beneficio anual [7] | 7 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de buena voluntad [5] | 5 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Clara infracción [5] | 5 |
| Detección de intrusiones | Detección activa en la aplicación [1] | 1 | Violación de privacidad | Cientos de personas [5] | 5 |

| Puntuación de probabilidad: | 3.875 | Puntuación de impacto: | 6.25 |
|---|---|---|---|

Severidad general del riesgo: **Alto**

| Probabilidad \ Impacto | Bajo | Medio | ->High<- |
|---|---|---|---|
| Bajo | Ninguno | Bajo | Medio |
| ->Medio<- | Bajo | Medio | ->Alto<- |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 44, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 44.**

*Prueba de la debilidad de la caché del navegador (OTG-AUTHN-006)*

**Evaluación de riesgos de OWASP**
**Prueba de la debilidad de la caché del navegador (OTG-AUTHN-006)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Sin conocimientos técnicos [1] | 1 | Pérdida de confidencialidad | Se divulgan numerosos datos críticos [7] | 7 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | No se requiere acceso ni recursos [9] | 9 | Pérdida de disponibilidad | Se interrumpieron amplios servicios primarios [7] | 7 |
| Tamaño de la poblacion | No aplica [0] | 0 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de cuentas importantes [4] | 4 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | Cientos de personas [5] | 5 |

| Puntuación de probabilidad: | 2.25 | Puntuación de impacto: | 3.375 |
|---|---|---|---|

Severidad general del riesgo: **Bajo**

| Probabilidad | Impacto: Bajo | ->Medio<- | Alto |
|---|---|---|---|
| ->Bajo<- | Ninguno | ->Bajo<- | Medio |
| Medio | Bajo | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 45, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 45.**

*Prueba de la política de contraseñas débiles (OTG-AUTHN-007)*

**Evaluación de riesgos de OWASP**
**Prueba de la política de contraseñas débiles (OTG-AUTHN-007)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Sin conocimientos técnicos [1] | 1 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de cuentas importantes [4] | 4 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | Detección activa en la aplicación [1] | 1 | Violación de privacidad | Un individuo [3] | 3 |

| Puntuación de probabilidad: | 1.875 | Puntuación de impacto: | 2.25 |
|---|---|---|---|

Severidad general del riesgo: **Ninguno**

| Probabilidad | Impacto: ->Bajo<- | Medio | Alto |
|---|---|---|---|
| ->Bajo<- | ->Ninguno<- | Bajo | Medio |
| Medio | Bajo | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 46, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 46.**

*Prueba de pregunta / respuesta de seguridad débil (OTG-AUTHN-008)*

**Evaluación de riesgos de OWASP**
**Prueba de pregunta / respuesta de seguridad débil (OTG-AUTHN-008)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Habilidades de penetración de la seguridad [9] | 9 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | Servicios secundarios mínimos interrumpidos [1] | 1 |
| Tamaño de la poblacion | No aplica [0] | 0 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de cuentas importantes [4] | 4 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | Registrado sin revisión [8] | 8 | Violación de privacidad | Un individuo [3] | 3 |
| **Puntuación de probabilidad:** | **3.5** | | **Puntuación de impacto:** | **2.5** | |

**Severidad general del riesgo:** **Bajo**

| **Probabilidad** | **Impacto** ->Bajo<- | Medio | Alto |
|---|---|---|---|
| Bajo | Ninguno | **Bajo** | Medio |
| ->Medio<- | ->Bajo<- | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 47, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 47.**

*Prueba de funcionalidades débiles de cambio o restablecimiento de contraseña (OTG-AUTHN-009)*

**Evaluación de riesgos de OWASP**
**Prueba de funcionalidades débiles de cambio o restablecimiento de contraseña (OTG-AUTHN-009)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | No aplica [0] | 0 | Pérdida de confidencialidad | No aplica [0] | 0 |
| Motivo | No aplica [0] | 0 | Pérdida de integridad | No aplica [0] | 0 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | No aplica [0] | 0 |
| Tamaño de la poblacion | No aplica [0] | 0 | Pérdida de responsabilidad | No aplica [0] | 0 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | No aplica [0] | 0 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | No aplica [0] | 0 | Daño a la reputación | No aplica [0] | 0 |
| Conciencia | No aplica [0] | 0 | Incumplimiento | No aplica [0] | 0 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | No aplica [0] | 0 |
| **Puntuación de probabilidad:** | **0** | | **Puntuación de impacto:** | **0** | |

**Severidad general del riesgo:** Ninguno

| **Probabilidad** | **Impacto** ->Bajo<- | Medio | Alto |
|---|---|---|---|
| ->Bajo<- | ->Ninguno<- | **Bajo** | Medio |
| Medio | **Bajo** | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 48, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 48.**

*Prueba de autenticación más débil en canal alternativo (OTG-AUTHN-010)*

**Evaluación de riesgos de OWASP**
**Prueba de autenticación más débil en canal alternativo (OTG-AUTHN-010)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | Registrado sin revisión [8] | 8 | Violación de privacidad | Un individuo [3] | 3 |

| Puntuación de probabilidad: | 3.375 | Puntuación de impacto: | 2 |
|---|---|---|---|

Severidad general del riesgo: **Bajo**

| Probabilidad | Impacto ->Bajo<- | Medio | Alto |
|---|---|---|---|
| Bajo | Ninguno | Bajo | Medio |
| ->Medio<- | ->Bajo<- | Medio | Alto |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

### 3.2.5 Pruebas de Autorización

A continuación, en la Tabla 49, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 49.**

*Prueba de inclusión de archivos / recorrido de directorio (OTG-AUTHZ-001)*

**Evaluación de riesgos de OWASP**
**Prueba de inclusión de archivos / recorrido de directorio (OTG-AUTHZ-001)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Sin conocimientos técnicos [1] | 1 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Servicios secundarios mínimos interrumpidos [1] | 1 |
| Tamaño de la poblacion | Usuarios de intranet [4] | 4 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | Registrado y revisado [3] | 3 | Violación de privacidad | Un individuo [3] | 3 |

| Puntuación de probabilidad: | 3.625 | Puntuación de impacto: | 1.5 |
|---|---|---|---|

Severidad general del riesgo: **Bajo**

| Probabilidad | Impacto ->Bajo<- | Medio | Alto |
|---|---|---|---|
| Bajo | Ninguno | Bajo | Medio |
| ->Medio<- | ->Bajo<- | Medio | Alto |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 50, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 50.**

*Prueba para eludir el esquema de autorización (OTG-AUTHZ-002)*

**Evaluación de riesgos de OWASP**
**Prueba para eludir el esquema de autorización (OTG-AUTHZ-002)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | No aplica [0] | 0 | Pérdida de confidencialidad | No aplica [0] | 0 |
| Motivo | No aplica [0] | 0 | Pérdida de integridad | No aplica [0] | 0 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | No aplica [0] | 0 |
| Tamaño de la poblacion | No aplica [0] | 0 | Pérdida de responsabilidad | No aplica [0] | 0 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | No aplica [0] | 0 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | No aplica [0] | 0 | Daño a la reputación | No aplica [0] | 0 |
| Conciencia | No aplica [0] | 0 | Incumplimiento | No aplica [0] | 0 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | No aplica [0] | 0 |
| **Puntuación de probabilidad:** | **0** | | **Puntuación de impacto:** | **0** | |

**Severidad general del riesgo:** Ninguno

| Probabilidad | Impacto ->Bajo<- | Medio | Alto |
|---|---|---|---|
| ->Bajo<- | ->Ninguno<- | Bajo | Medio |
| Medio | Bajo | Medio | Alto |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 51, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor medio en la severidad de riesgo.

**Tabla 51.**

*Prueba de escalamiento de privilegios (OTG-AUTHZ-003)*

**Evaluación de riesgos de OWASP**
**Prueba de escalamiento de privilegios (OTG-AUTHZ-003)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | Efecto significativo en el beneficio anual [7] | 7 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de cuentas importantes [4] | 4 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | Clara infracción [5] | 5 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | Cientos de personas [5] | 5 |
| **Puntuación de probabilidad:** | **4.75** | | **Puntuación de impacto:** | **4.75** | |

**Severidad general del riesgo:** Medio

| Probabilidad | Impacto Bajo | ->Medio<- | Alto |
|---|---|---|---|
| Bajo | Ninguno | Bajo | Medio |
| ->Medio<- | Bajo | ->Medio<- | Alto |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 52, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor alto en la severidad de riesgo.

**Tabla 52.**

*Prueba de referencias de objetos directos inseguros (OTG-AUTHZ-004)*

**Evaluación de riesgos de OWASP**
**Prueba de referencias de objetos directos inseguros (OTG-AUTHZ-004)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Habilidades de penetración de la seguridad [9] | 9 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Numerosos datos muy corruptos [7] | 7 |
| Oportunidad | No se requiere acceso ni recursos [9] | 9 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Fácil [7] | 7 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de buena voluntad [5] | 5 |
| Conciencia | Obvio [6] | 6 | Incumplimiento | Violación de alto perfil [7] | 7 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | Cientos de personas [5] | 5 |

| **Puntuación de probabilidad:** | **6.625** | **Puntuación de impacto:** | **4.875** |
|---|---|---|---|

**Severidad general del riesgo:** **Alto**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | Bajo | **->Medio<-** | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| Medio | **Bajo** | Medio | Alto |
| **->Alto<-** | Medio | **->Alto<-** | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

### 3.2.6 Pruebas de gestión de sesiones

A continuación, en la Tabla 53, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 53.**

*Prueba para omitir el esquema de administración de sesiones (OTG-SESS-001)*

**Evaluación de riesgos de OWASP**
**Prueba para omitir el esquema de administración de sesiones (OTG-SESS-001)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | No aplica [0] | 0 |

| **Puntuación de probabilidad:** | **4.75** | **Puntuación de impacto:** | **1.625** |
|---|---|---|---|

**Severidad general del riesgo:** **Bajo**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | **->Bajo<-** | Medio | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| **->Medio<-** | **->Bajo<-** | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 54, se detalla el valor de la evaluación de riego de la prueba realizada en la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 54.**

*Prueba de fijación de sesión (OTG-SESS-003)*

**Evaluación de riesgos de OWASP**
**Prueba de fijación de sesión (OTG-SESS-003)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | No aplica [0] | 0 |

| **Puntuación de probabilidad:** | **3.875** | **Puntuación de impacto:** | **2.375** |
|---|---|---|---|

**Severidad general del riesgo:** **Bajo**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | **->Bajo<-** | Medio | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| **->Medio<-** | **->Bajo<-** | Medio | Alto |
| Alto | Medio | Alto | Critico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 55, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor medio en la severidad de riesgo.

**Tabla 55.**

*Prueba de falsificación de solicitudes entre sitios (CSRF) (OTG-SESS-005)*

**Evaluación de riesgos de OWASP**
**Prueba de falsificación de solicitudes entre sitios (CSRF) (OTG-SESS-005)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Habilidades de penetración de la seguridad [9] | 9 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos extensos ligeramente corruptos [5] | 5 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Se interrumpieron amplios servicios primarios [7] | 7 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Obvio [6] | 6 | Incumplimiento | No aplica [0] | 0 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | No aplica [0] | 0 |

| **Puntuación de probabilidad:** | **5.875** | **Puntuación de impacto:** | **3.375** |
|---|---|---|---|

**Severidad general del riesgo:** **Medio**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | Bajo | **->Medio<-** | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| **->Medio<-** | **Bajo** | **->Medio<-** | Alto |
| Alto | Medio | Alto | Critico |

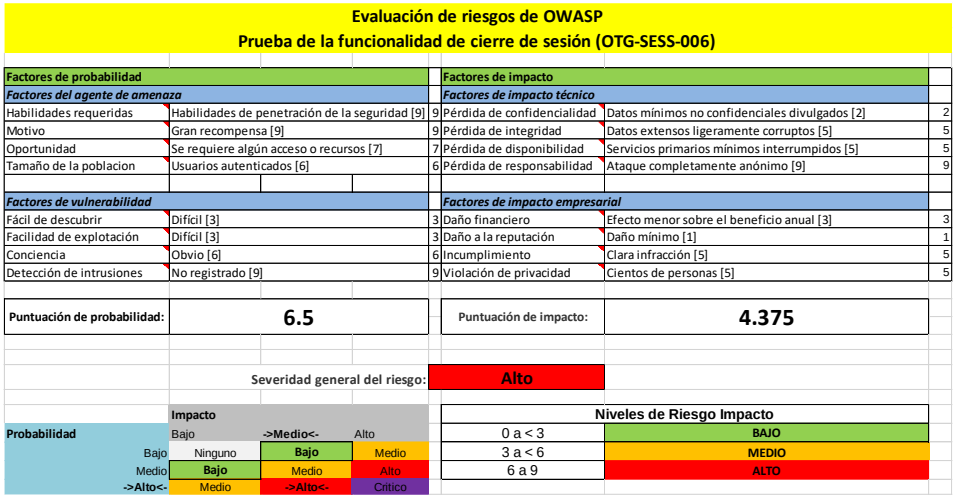
| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 56, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor alto en la severidad de riesgo.

**Tabla 56.**

*Prueba de la funcionalidad de cierre de sesión (OTG-SESS-006)*



**Evaluación de riesgos de OWASP**
**Prueba de la funcionalidad de cierre de sesión (OTG-SESS-006)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Habilidades de penetración de la seguridad [9] | 9 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Gran recompensa [9] | 9 | Pérdida de integridad | Datos extensos ligeramente corruptos [5] | 5 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque completamente anónimo [9] | 9 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Obvio [6] | 6 | Incumplimiento | Clara infracción [5] | 5 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | Cientos de personas [5] | 5 |

| Puntuación de probabilidad: | 6.5 | Puntuación de impacto: | 4.375 |
|---|---|---|---|

Severidad general del riesgo: **Alto**

| Probabilidad \ Impacto | Bajo | ->Medio<- | Alto |
|---|---|---|---|
| Bajo | Ninguno | Bajo | Medio |
| Medio | Bajo | Medio | Alto |
| ->Alto<- | Medio | ->Alto<- | Crítico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

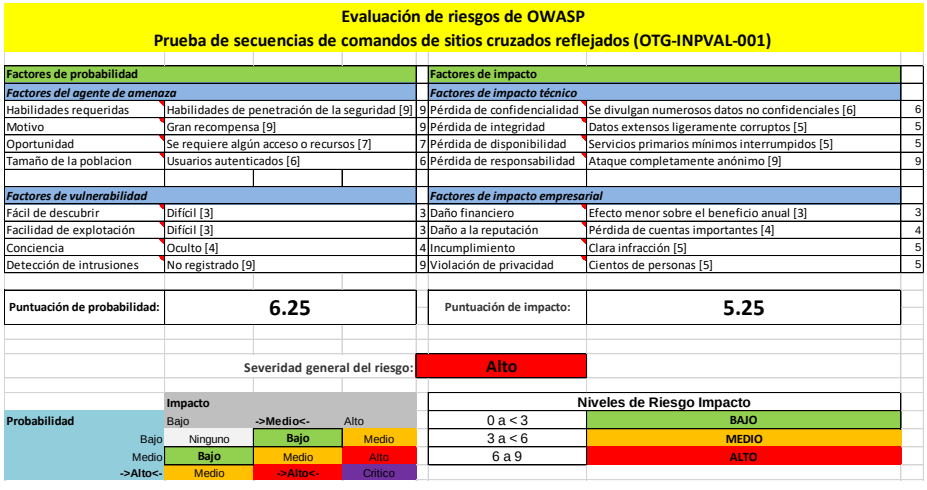
Fuente: elaboración propia

### 3.2.7 Pruebas de validación de entradas

A continuación, en la Tabla 57, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor alto en la severidad de riesgo.

**Tabla 57.**

*Prueba de secuencias de comandos de sitios cruzados reflejados (OTG-INPVAL-001)*



**Evaluación de riesgos de OWASP**
**Prueba de secuencias de comandos de sitios cruzados reflejados (OTG-INPVAL-001)**

| Factores de probabilidad | | | Factores de impacto | | |
|---|---|---|---|---|---|
| *Factores del agente de amenaza* | | | *Factores de impacto técnico* | | |
| Habilidades requeridas | Habilidades de penetración de la seguridad [9] | 9 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Gran recompensa [9] | 9 | Pérdida de integridad | Datos extensos ligeramente corruptos [5] | 5 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque completamente anónimo [9] | 9 |
| *Factores de vulnerabilidad* | | | *Factores de impacto empresarial* | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de cuentas importantes [4] | 4 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Clara infracción [5] | 5 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | Cientos de personas [5] | 5 |

| Puntuación de probabilidad: | 6.25 | Puntuación de impacto: | 5.25 |
|---|---|---|---|

Severidad general del riesgo: **Alto**

| Probabilidad \ Impacto | Bajo | ->Medio<- | Alto |
|---|---|---|---|
| Bajo | Ninguno | Bajo | Medio |
| Medio | Bajo | Medio | Alto |
| ->Alto<- | Medio | ->Alto<- | Crítico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 58, se detalla el valor de la evaluación de riego de la prueba realizada, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 58.**

*Prueba de manipulación verbos (OTG-INPVAL-003)*

| Evaluación de riesgos de OWASP<br>Prueba de manipulación verbos (OTG-INPVAL-003) | | | | | |
|---|---|---|---|---|---|
| **Factores de probabilidad** | | | **Factores de impacto** | | |
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | No aplica [0] | 0 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos levemente corruptos [1] | 1 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de cuentas importantes [4] | 4 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | No aplica [0] | 0 |
| **Puntuación de probabilidad:** | **1.25** | | **Puntuación de impacto:** | **2** | |
| | | | **Severidad general del riesgo:** | Ninguno | |

| Probabilidad | Impacto ->Bajo<- | Medio | Alto |
|---|---|---|---|
| ->Bajo<- | ->Ninguno<- | Bajo | Medio |
| Medio | Bajo | Medio | Alto |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 59, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor medio en la severidad de riesgo.

**Tabla 59.**

*Prueba de contaminación de parámetros HTTP (OTG-INPVAL-004)*

| Evaluación de riesgos de OWASP<br>Prueba de contaminación de parámetros HTTP (OTG-INPVAL-004) | | | | | |
|---|---|---|---|---|---|
| **Factores de probabilidad** | | | **Factores de impacto** | | |
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Datos extensos ligeramente corruptos [5] | 5 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de cuentas importantes [4] | 4 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | Cientos de personas [5] | 5 |
| **Puntuación de probabilidad:** | **4.75** | | **Puntuación de impacto:** | **3.875** | |
| | | | **Severidad general del riesgo:** | Medio | |

| Probabilidad | Impacto Bajo | ->Medio<- | Alto |
|---|---|---|---|
| Bajo | Ninguno | Bajo | Medio |
| ->Medio<- | Bajo | ->Medio<- | Alto |
| Alto | Medio | Alto | Critico |

| Niveles de Riesgo Impacto | |
|---|---|
| 0 a < 3 | BAJO |
| 3 a < 6 | MEDIO |
| 6 a 9 | ALTO |

Fuente: elaboración propia

A continuación, en la Tabla 60, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor alto en la severidad de riesgo.

**Tabla 60.**

*Prueba de inyección SQL (OTG-INPVAL-005)*

**Evaluación de riesgos de OWASP**
**Prueba de inyección SQL (OTG-INPVAL-005)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Habilidades de penetración de la seguridad [9] | 9 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Gran recompensa [9] | 9 | Pérdida de integridad | Numerosos datos muy corruptos [7] | 7 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Usuarios de Internet anónimos [9] | 9 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | El daño cuesta menos que solucionar el problema [1] | 1 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de buena voluntad [5] | 5 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Clara infracción [5] | 5 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | Cientos de personas [5] | 5 |

| **Puntuación de probabilidad:** | **6.625** | **Puntuación de impacto:** | **5.125** |
|---|---|---|---|

**Severidad general del riesgo:** **Alto**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | Bajo | **->Medio<-** | Alto |
| Bajo | Ninguno | **Bajo** | Medio |
| Medio | **Bajo** | Medio | Alto |
| **->Alto<-** | Medio | **->Alto<-** | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 61, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor bajo en la severidad de riesgo.

**Tabla 61.**

*Prueba de inyección de comando (OTG-INPVAL-013)*

**Evaluación de riesgos de OWASP**
**Prueba de inyección de comando (OTG-INPVAL-013)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | No aplica [0] | 0 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | Se requiere acceso completo o recursos | 0 | Pérdida de disponibilidad | Servicios primarios mínimos interrumpidos [5] | 5 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de cuentas importantes [4] | 4 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | Detección activa en la aplicación [1] | 1 | Violación de privacidad | Un individuo [3] | 3 |

| **Puntuación de probabilidad:** | **1.375** | **Puntuación de impacto:** | **3.375** |
|---|---|---|---|

**Severidad general del riesgo:** **Bajo**

| | **Impacto** | | |
|---|---|---|---|
| **Probabilidad** | Bajo | **->Medio<-** | Alto |
| **->Bajo<-** | Ninguno | **->Bajo<-** | Medio |
| Medio | **Bajo** | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

### 3.2.8 Pruebas de manejo de errores

A continuación, en la Tabla 62, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor de ninguno en la severidad de riesgo.

**Tabla 62.**

*Análisis de códigos de error (OTG-ERR-001)*

**Evaluación de riesgos de OWASP**
**Análisis de códigos de error (OTG-ERR-001)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Algunas habilidades técnicas [3] | 3 | Pérdida de confidencialidad | Datos mínimos no confidenciales divulgados [2] | 2 |
| Motivo | Recompensa baja o nula [1] | 1 | Pérdida de integridad | Datos mínimos muy corruptos [3] | 3 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | Servicios secundarios mínimos interrumpidos [1] | 1 |
| Tamaño de la poblacion | Administradores del sistema [2] | 2 | Pérdida de responsabilidad | Ataque completamente rastreable hasta el individuo [1] | 1 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | No aplica [0] | 0 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Daño mínimo [1] | 1 |
| Conciencia | Desconocido [1] | 1 | Incumplimiento | Infracción menor [2] | 2 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | No aplica [0] | 0 |
| **Puntuación de probabilidad:** | **2.125** | | **Puntuación de impacto:** | **1.25** | |

**Severidad general del riesgo:** Ninguno

| **Probabilidad** | **Impacto** ->Bajo<- | Medio | Alto |
|---|---|---|---|
| ->Bajo<- | ->Ninguno<- | **Bajo** | Medio |
| Medio | **Bajo** | Medio | Alto |
| Alto | Medio | Alto | Crítico |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

### 3.2.9 Pruebas de criptografía débil

A continuación, en la Tabla 63, se detalla el valor de la evaluación de riego de la prueba realizada, la cual, se determina un valor critico en la severidad de riesgo.

**Tabla 63.**

*Prueba de cifrados SSL / TLS débiles, protección insuficiente de la capa de transporte (OTG-CRYPST-001)*

**Evaluación de riesgos de OWASP**
**Prueba de cifrados SSL / TLS débiles, protección insuficiente de la capa de transporte (OTG-CRYPST-001)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Habilidades de programación y redes [6] | 6 | Pérdida de confidencialidad | Se divulgan numerosos datos críticos [7] | 7 |
| Motivo | Gran recompensa [9] | 9 | Pérdida de integridad | Numerosos datos muy corruptos [7] | 7 |
| Oportunidad | Se requiere algún acceso o recursos [7] | 7 | Pérdida de disponibilidad | Todos los servicios completamente perdidos [9] | 9 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Herramientas automatizadas disponibles [9] | 9 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Fácil [5] | 5 | Daño a la reputación | Pérdida de buena voluntad [5] | 5 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Clara infracción [5] | 5 |
| Detección de intrusiones | No registrado [9] | 9 | Violación de privacidad | Cientos de personas [5] | 5 |
| **Puntuación de probabilidad:** | **6.875** | | **Puntuación de impacto:** | **6** | |

**Severidad general del riesgo:** Critico

| **Probabilidad** | **Impacto** Bajo | Medio | ->High<- |
|---|---|---|---|
| Bajo | Ninguno | **Bajo** | Medio |
| Medio | **Bajo** | Medio | Alto |
| ->Alto<- | Medio | Alto | ->Critico<- |

| **Niveles de Riesgo Impacto** | |
|---|---|
| 0 a < 3 | **BAJO** |
| 3 a < 6 | **MEDIO** |
| 6 a 9 | **ALTO** |

Fuente: elaboración propia

A continuación, en la Tabla 64, se detalla el valor de la evaluación de riego de la prueba realizada en, la cual, se determina un valor medio en la severidad de riesgo.

**Tabla 64.**

*Prueba de información confidencial enviada a través de canales no cifrados (OTG-CRYPST-003)*

**Evaluación de riesgos de OWASP**
**Prueba de información confidencial enviada a través de canales no cifrados (OTG-CRYPST-003)**

| **Factores de probabilidad** | | | **Factores de impacto** | | |
|---|---|---|---|---|---|
| ***Factores del agente de amenaza*** | | | ***Factores de impacto técnico*** | | |
| Habilidades requeridas | Usuario de computadora avanzado [5] | 5 | Pérdida de confidencialidad | Se divulgan numerosos datos no confidenciales [6] | 6 |
| Motivo | Posible recompensa [4] | 4 | Pérdida de integridad | Numerosos datos muy corruptos [7] | 7 |
| Oportunidad | Se requiere acceso o recursos especiales [4] | 4 | Pérdida de disponibilidad | Todos los servicios completamente perdidos [9] | 9 |
| Tamaño de la poblacion | Usuarios autenticados [6] | 6 | Pérdida de responsabilidad | Ataque posiblemente rastreable hasta un individuo [7] | 7 |
| ***Factores de vulnerabilidad*** | | | ***Factores de impacto empresarial*** | | |
| Fácil de descubrir | Difícil [3] | 3 | Daño financiero | Efecto menor sobre el beneficio anual [3] | 3 |
| Facilidad de explotación | Difícil [3] | 3 | Daño a la reputación | Pérdida de buena voluntad [5] | 5 |
| Conciencia | Oculto [4] | 4 | Incumplimiento | Clara infracción [5] | 5 |
| Detección de intrusiones | No aplica [0] | 0 | Violación de privacidad | Cientos de personas [5] | 5 |

| **Puntuación de probabilidad:** | **3.625** | **Puntuación de impacto:** | **5.875** |
|---|---|---|---|

**Severidad general del riesgo:** **Medio**

| | **Impacto** | | | **Niveles de Riesgo Impacto** | |
|---|---|---|---|---|---|
| **Probabilidad** | Bajo | **->Medio<-** | Alto | 0 a < 3 | **BAJO** |
| Bajo | Ninguno | **Bajo** | Medio | 3 a < 6 | **MEDIO** |
| **->Medio<-** | **Bajo** | **->Medio<-** | Alto | 6 a 9 | **ALTO** |
| Alto | Medio | Alto | Crítico | | |

Fuente: elaboración propia

## 3.3 Análisis de Resultados

El resumen de los resultados encontrados que se aplica mediante la metodología OWASP, se muestran a continuación, se toma en cuenta que en cada prueba se realizó el respectivo análisis de la información encontrada, toda la información encontrada y detallada en la presente investigación es de mucha importancia, se consigue tomar alguna decisión para tomar alguna medida para solucionar o tratar de mitigar posibles vulnerabilidades dentro de las aplicaciones web de la Institución.

En esta sección, se presenta los resultados obtenidos y previamente analizados, de las guías de la metodología OWASP v4.0.

### 3.3.1 Recopilación de Información

A continuación, en la Tabla 65, se muestran los resultados obtenidos de la recopilación de la información.

**Tabla 65.**

*Recopilación de información*

| CATEGORÍA | CÓDIGO DE PRUEBA | RESULTADO DE LA PRUEBA |
|---|---|---|
| Recopilación de la información | OTG-INFO-001 | En el resultado de la prueba, se obtiene datos de subdominios, se encuentra tecnologías web, puertos abiertos, servidor web, y certificado |
| | OTG-INFO-002 | Se encuentra información acerca del servidor Apache, el cual, muestra que el servidor esta levantado y en funcionamiento, además, la versión del servidor |
| | OTG-INFO-003 | Al analizar las pruebas realizadas no se encuentra archivos maliciosos, como el archivo robots.txt, y tampoco se encuentra páginas de administración de la aplicación Koha |
| | OTG-INFO-004 | En esta prueba, se lista algunos puertos que se ejecutan, es decir, que esta abiertos, así también, de los servicios y versión que corresponden a cada puerto abierto |
| | OTG-INFO-005 | No se encuentran metadatos, que indiquen o muestren alguna información importante de la aplicación web, por tanto, no se encuentra información sensible |
| | OTG-INFO-006 | Se encuentra información importante y confidencial como contraseñas mediante el método post, y enlaces de otras páginas indican los códigos con los que se envían las peticiones al servidor web, las cuales se logran determinar que son de fácil vulnerabilidad |
| | OTG-INFO-007 | Se determina que existen mapas de directorios los cuales indican archivos JavaScript, los mismos que tiene algún tipo de seguridad |
| | OTG-INFO-008 | El resultado retorno que se muestra la información del servidor web y la respectiva versión de este |
| | OTG-INFO-009 | El resultado muestra que se ejecutan plugin a través del framework Drupal, y de la misma manera mediante de WordPress, lo cual, indica que es información sensible |

Fuente: elaboración propia

### 3.3.2 Pruebas de gestión de la configuración y la implementación

A continuación, en la Tabla 66, se muestran los resultados obtenidos de las pruebas de gestión de la configuración y la implementación.

**Tabla 66.**

*Pruebas de gestión de la configuración*

| CATEGORÍA | CÓDIGO DE PRUEBA | RESULTADO DE LA PRUEBA |
|---|---|---|
| Pruebas de gestión de la configuración y la implementación | OTG-CONFIG-001 | Se muestra el detalle de la Infraestructura de la red de la Institución, con la finalidad de determinar el nivel de seguridad en los puntos finales de la red, debido a la complejidad de la red institucional únicamente se obtiene una parte de la infraestructura |
| | OTG-CONFIG-002 | El resultado de la prueba indica que existen módulos instalados en el servidor web, los cuales pertenecen al Framework Koha. |
| | OTG-CONFIG-003 | Se indica las peticiones realizadas por otras aplicaciones hacia al servidor web, las cuales se activan para cumplir el correcto funcionamiento del servidor, además, contienen información vulnerable |
| | OTG-CONFIG-004 | Se obtiene información sobre los directorios dentro del servidor que pertenece a la aplicación web, los cuales no contiene archivos viejos, o respaldos anteriores |
| | OTG-CONFIG-005 | En esta prueba, se indica la única interfaz administrativa, por parte de la aplicación web, y emitida por el framework Koha. |
| | OTG-CONFIG-006 | Se indica los métodos que se encuentran en ejecución en el servidor por parte de GET y POST, y la habilitación de TRACE |
| | OTG-CONFIG-007 | Los resultados determinan que no se encuentra implementado el mecanismo de seguridad HSTS |
| | OTG-CONFIG-008 | Los resultados determinan que no se encuentra implementado el mecanismo de seguridad contra la vulnerabilidad XSS Cross-Site |

Fuente: elaboración propia

### 3.3.3 Pruebas de gestión de identidad

A continuación, en la Tabla 67, se muestran los resultados obtenidos de las pruebas de gestión de identidad.

**Tabla 67.**

*Pruebas de gestión de identidad*

| CATEGORÍA | CÓDIGO DE PRUEBA | RESULTADO DE LA PRUEBA |
|---|---|---|
| Pruebas de gestión de identidad | OTG-IDENT-001 | Se determino que existen varios roles que hacen uso de la aplicación web, para otorgar permisos y a su vez ser usuarios finales de la aplicación |
| | OTG-IDENT-002 | Se plantearon varias preguntas en cuanto al proceso y manejo de la creación de usuarios para la aplicación web |
| | OTG-IDENT-003 | Verificar cuentas asociadas a la aplicación web, las cuales fueron únicamente pertenecientes a la misma |

Fuente: elaboración propia

### 3.3.4 Pruebas de Autenticación

A continuación, en la Tabla 68, se muestran los resultados obtenidos de las pruebas de autenticación.

**Tabla 68.**

*Pruebas de Autenticación*

| CATEGORÍA | CÓDIGO DE PRUEBA | RESULTADO DE LA PRUEBA |
|---|---|---|
| Pruebas de Autenticación | OTG-AUTHN-001 | El resultado de esta prueba indica que existe información sensible en cuanto a la entrega de credenciales de autenticación al acceder a la aplicación web. |
| | OTG-AUTHN-002 | Para realizar esta prueba se requiere una herramienta de fuerza bruta, pero no se la realiza, no se obtiene los recursos necesarios para implementarla. |
| | OTG-AUTHN-003 | El resultado muestra los mecanismos de seguridad para el acceso a la aplicación web, tanto como el bloqueo y las preguntas de seguridad. |

| | | |
|---|---|---|
| | OTG-AUTHN-004 | La prueba indica la redirección de la página principal de la aplicación a una nueva url, la cual, indica el registro de sesión del usuario ingresado |
| | OTG-AUTHN-005 | Se logra verificar en esta prueba las cookies almacenadas al ingresar un usuario a la aplicación web. |
| | OTG-AUTHN-006 | Se logra determinar que la cache en el navegador, se almacena, lo cual, consigue ser perjudicial para la aplicación, se logra obtener datos confidenciales a través de la cache de la aplicación web |
| | OTG-AUTHN-007 | No se encuentra flujos de datos para adivinar fácilmente la contraseña para ingresar a la aplicación web |
| | OTG-AUTHN-008 | La aplicación web si cuenta con la respectiva seguridad para la recuperación de la contraseña, no se la realiza a través de la aplicación, sino mediante un envío de correo electrónico a través de la misma aplicación |
| | OTG-AUTHN-009 | No se logra determinar esta prueba debido a que la recuperación de contraseña a través de la aplicación no está contemplada |
| | OTG-AUTHN-010 | Se determina que, para la recuperación de contraseña, se las realiza a través de los métodos de dispositivos móviles y a través de la página web |

Fuente: elaboración propia

### 3.3.5 Pruebas de Autorización

A continuación, en la Tabla 69, se muestran los resultados obtenidos de las pruebas de autorización.

**Tabla 69.**

*Pruebas de Autorización*

| CATEGORÍA | CÓDIGO DE PRUEBA | RESULTADO DE LA PRUEBA |
|---|---|---|
| Pruebas de Autorización | OTG-AUTHZ-001 | El resultado de la prueba indica que si esta la seguridad respectiva para la autorización de permisos de directorios, es decir, un análisis transversal |

| | | |
|---|---|---|
| | OTG-AUTHZ-002 | No se alcanza a realizar esta prueba debido a que no se tiene acceso completo a todos los directorios de la aplicación web |
| | OTG-AUTHZ-003 | Se consigue identificar mediante esta prueba que se filtra los identificadores de cada usuario al ingresar a la aplicación web. |
| | OTG-AUTHZ-004 | El resultado de esta prueba determina que se alcanza fácilmente modificar los parámetros de las paginas, las cuales envían mediante métodos los valores a través de la url |

Fuente: elaboración propia

### 3.3.6 Pruebas de gestión de sesiones

A continuación, en la Tabla 70, se muestran los resultados obtenidos de las pruebas de gestión de sesiones.

**Tabla 70.**

*Pruebas de gestión de sesiones*

| CATEGORÍA | CÓDIGO DE PRUEBA | RESULTADO DE LA PRUEBA |
|---|---|---|
| Prueba de gestión de sesiones | OTG-SESS-001 | La prueba identifica que existe cookies que se ejecutan dentro de la aplicación web, posterior al ingreso de un usuario al sistema |
| | OTG-SESS-003 | Se determina mediante el resultado de la prueba que se logra interceptar los datos de las cookies |
| | OTG-SESS-005 | El resultado de la prueba determina que, se alcanza a realizar una intercepción de la información a través de la vulnerabilidad de Cross-site request forgery |
| | OTG-SESS-006 | Se encuentra vulnerabilidades conocidas que responden a Cross Site Scripting y Cross Site Request Forgery, las cuales alcanzan a ser interceptados por un atacante |

Fuente: elaboración propia

### 3.3.7 Pruebas de validación de entradas

A continuación, en la Tabla 71, se muestran los resultados obtenidos de las pruebas de validación de entradas.

**Tabla 71.**

*Pruebas de validación de entradas*

| CATEGORÍA | CÓDIGO DE PRUEBA | RESULTADO DE LA PRUEBA |
|---|---|---|
| Prueba de validación de entrada | OTG-INPVAL-001 | El resultado de esta prueba determina que, se consigue realizar la intercepción de la contraseña de un usuario, mediante el ataque de XSS (Cross Site Scripting) |
| | OTG-INPVAL-003 | No se encuentra métodos se ejecutan en la aplicación, es decir, no se muestran directamente en la aplicación mediante el método http |
| | OTG-INPVAL-004 | Se obtiene métodos que devuelven una url con parámetros que se envían a través de la ejecución de alguna función dentro de la aplicación, como valores de identificadores |
| | OTG-INPVAL-005 | La prueba que se realiza indica las vulnerabilidades de la aplicación web, ya que, logra estar sujeta a ser atacada por un ciberdelincuente, a través de inyección SQL |
| | OTG-INPVAL-013 | No se obtiene información sensible inyectando código al sistema operativo mediante una solicitud HTTP |

Fuente: elaboración propia

### 3.3.8 Pruebas de manejo de errores

A continuación, en la Tabla 72 se muestran los resultados obtenidos de las pruebas de manejo de errores.

**Tabla 72.**

*Pruebas de manejos de errores*

| CATEGORÍA | CÓDIGO DE PRUEBA | RESULTADO DE LA PRUEBA |
|---|---|---|
| Prueba de manejo de errores | OTG-ERR-001 | Se determina mediante esta prueba, que el manejo de errores se encuentra configurado, y se presenta un mensaje de error ante alguna respuesta errónea del sistema |

Fuente: elaboración propia

### 3.3.9 Pruebas de Criptografía débil

A continuación, en la Tabla 73, se muestran los resultados obtenidos de las pruebas de criptografía débil.

**Tabla 73.**

*Pruebas de criptografía débil*

| CATEGORÍA | CÓDIGO DE PRUEBA | RESULTADO DE LA PRUEBA |
|---|---|---|
| Pruebas de Criptografía débil | OTG-CRYPST-001 | El resultado de la prueba indica que, no se cuenta con un certificado TLS/SSL, el cual, es importante para mantener segura la aplicación ante alguna amenaza por un atacante. |
| | OTG-CRYPST-003 | Se determina mediante esta prueba que la información de la aplicación no se encuentra cifrada, únicamente se codifica y se envían como encabezado de HTTP |

Fuente: elaboración propia

Finalmente, se muestra los resultados de las diferentes pruebas realizadas, un resumen que indica los valores de riesgo por cada prueba realizada a las aplicaciones web de la institución.

En la Tabla 74, se muestra los resultados consolidados de las diferentes pruebas realizadas a la aplicación web, con el respectivo nivel de riego de cada una de ellas, se toma en cuenta la Guía de Pruebas de OWASP v4.0.

**Tabla 74.**

*Checklist Pruebas OWASP v4.0*

| Categoría | ID de prueba | Categoria | Nombre de la prueba | Pruebas | Riesgo | A | M | B | N | N/A |
|---|---|---|---|---|---|---|---|---|---|---|
| Recopilación de información | OTG-INFO-001 | Configuración | Llevar a cabo el descubrimiento y el reconocimiento de motores de búsqueda para detectar fugas de información | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Recopilación de información | OTG-INFO-002 | Configuración | Servidor web de huellas digitales | Hecho | Medio | 0 | 1 | 0 | 0 | 0 |
| Recopilación de información | OTG-INFO-003 | Configuración | Revisar los metarchivos del servidor web para detectar fugas de información | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Recopilación de información | OTG-INFO-004 | Configuración | Enumerar aplicaciones en el servidor web | Hecho | Medio | 0 | 1 | 0 | 0 | 0 |
| Recopilación de información | OTG-INFO-005 | Calidad del código | Revisar los comentarios y metadatos de la página web para detectar fugas de información | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Recopilación de información | OTG-INFO-006 | Configuración | Identificar los puntos de entrada de la aplicación | Hecho | Alto | 1 | 0 | 0 | 0 | 0 |
| Recopilación de información | OTG-INFO-007 | Configuración | Mapear rutas de ejecución a través de la aplicación | Hecho | Medio | 0 | 1 | 0 | 0 | 0 |
| Recopilación de información | OTG-INFO-008 | Configuración | Marco de aplicación web de huellas dactilares | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Recopilación de información | OTG-INFO-009 | Configuración | Aplicación web de huellas dactilares | Hecho | Medio | 0 | 1 | 0 | 0 | 0 |
| Pruebas de gestión de configuración e implementación | OTG-CONFIG-001 | Configuración | Prueba de configuración de red / infraestructura | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Pruebas de gestión de configuración e implementación | OTG-CONFIG-002 | Configuración | Prueba de la configuración de la plataforma de aplicaciones | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Pruebas de gestión de configuración e implementación | OTG-CONFIG-003 | Manejo de errores | Manejo de extensiones de archivo de prueba para información confidencial | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Pruebas de gestión de configuración e implementación | OTG-CONFIG-004 | Ambiental | Archivos de respaldo y no referenciados para información confidencial | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Pruebas de gestión de configuración e implementación | OTG-CONFIG-005 | Configuración | Enumerar las interfaces de administración de aplicaciones e infraestructura | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Pruebas de gestión de configuración e implementación | OTG-CONFIG-006 | Configuración | Probar métodos HTTP | Hecho | Medio | 0 | 1 | 0 | 0 | 0 |
| Pruebas de gestión de configuración e implementación | OTG-CONFIG-007 | Configuración | Probar la seguridad de transporte estricta de HTTP | Hecho | Alto | 1 | 0 | 0 | 0 | 0 |
| Pruebas de gestión de configuración e implementación | OTG-CONFIG-008 | Configuración | Probar la política de dominios cruzados de RIA | Hecho | Alto | 1 | 0 | 0 | 0 | 0 |
| Pruebas de gestión de identidad | OTG-IDENT-001 | Autorización | Definiciones de roles de prueba | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Pruebas de gestión de identidad | OTG-IDENT-002 | Autenticación | Proceso de registro de usuario de prueba | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Pruebas de gestión de identidad | OTG-IDENT-003 | Autenticación | Probar el proceso de aprovisionamiento de cuentas | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Pruebas de gestión de identidad | OTG-IDENT-004 | - | Prueba de enumeración de cuentas y cuentas de usuario adivinables | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Pruebas de gestión de identidad | OTG-IDENT-005 | - | Prueba de la política de nombre de usuario débil o no impuesta | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Pruebas de gestión de identidad | OTG-IDENT-006 | - | Permisos de prueba de cuentas de invitado / formación | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Pruebas de gestión de identidad | OTG-IDENT-007 | - | Probar el proceso de suspensión / reanudación de la cuenta | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Prueba de autenticación | OTG-AUTHN-001 | Autenticación | Prueba de credenciales transportadas a través de un canal cifrado | Hecho | Alto | 1 | 0 | 0 | 0 | 0 |
| Prueba de autenticación | OTG-AUTHN-002 | Autenticación | Prueba de credenciales predeterminadas | Hecho | Medio | 0 | 1 | 0 | 0 | 0 |
| Prueba de autenticación | OTG-AUTHN-003 | Autenticación | Prueba de mecanismo de bloqueo débil | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Prueba de autenticación | OTG-AUTHN-004 | Autenticación | Prueba para omitir el esquema de autenticación | Hecho | Medio | 0 | 1 | 0 | 0 | 0 |
| Prueba de autenticación | OTG-AUTHN-005 | Autenticación | Prueba la funcionalidad de recordar contraseña | Hecho | Alto | 1 | 0 | 0 | 0 | 0 |
| Prueba de autenticación | OTG-AUTHN-006 | Configuración | Prueba de la debilidad de la caché del navegador | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Prueba de autenticación | OTG-AUTHN-007 | Autenticación | Prueba de la política de contraseñas débiles | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Prueba de autenticación | OTG-AUTHN-008 | Autenticación | Prueba de pregunta / respuesta de seguridad débil | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Prueba de autenticación | OTG-AUTHN-009 | Autenticación | Prueba de funciones débiles de cambio o restablecimiento de contraseña | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Prueba de autenticación | OTG-AUTHN-010 | Autenticación | Prueba de autenticación más débil en canal alternativo | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Prueba de autorización | OTG-AUTHZ-001 | Configuración | Prueba transversal de directorio / archivo incluido | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Prueba de autorización | OTG-AUTHZ-002 | Autorización | Prueba para omitir el esquema de autorización | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Prueba de autorización | OTG-AUTHZ-003 | Autorización | Prueba de escalamiento de privilegios | Hecho | Medio | 0 | 1 | 0 | 0 | 0 |
| Prueba de autorización | OTG-AUTHZ-004 | Autorización | Prueba de referencias de objetos directos inseguras | Hecho | Alto | 1 | 0 | 0 | 0 | 0 |
| Prueba de gestión de sesiones | OTG-SESS-001 | Autorización | Prueba del esquema de gestión de sesiones | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Prueba de gestión de sesiones | OTG-SESS-002 | - | Prueba de atributos de cookies | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Prueba de gestión de sesiones | OTG-SESS-003 | Autenticación | Prueba de fijación de sesión | Hecho | Bajo | 0 | 0 | 1 | 0 | 0 |
| Prueba de gestión de sesiones | OTG-SESS-004 | - | Prueba de variables de sesión expuestas | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Prueba de gestión de sesiones | OTG-SESS-005 | Gestión de sesiones | Prueba de falsificación de solicitudes entre sitios | Hecho | Medio | 0 | 1 | 0 | 0 | 0 |
| Prueba de gestión de sesiones | OTG-SESS-006 | Gestión de sesiones | Prueba de la funcionalidad de cierre de sesión | Hecho | Alto | 1 | 0 | 0 | 0 | 0 |
| Prueba de gestión de sesiones | OTG-SESS-007 | - | Tiempo de espera de la sesión de prueba | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Prueba de gestión de sesiones | OTG-SESS-008 | - | Prueba de la sesión desconcertante | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Pruebas de validación de datos | OTG-INPVAL-001 | Validación de entrada | Prueba de secuencias de comandos de sitios cruzados reflejados | Hecho | Alto | 1 | 0 | 0 | 0 | 0 |
| Pruebas de validación de datos | OTG-INPVAL-002 | - | Prueba de secuencias de comandos de sitios cruzados almacenadas | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Pruebas de validación de datos | OTG-INPVAL-003 | Validación de entrada | Prueba de manipulación de verbos HTTP | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Pruebas de validación de datos | OTG-INPVAL-004 | Validación de entrada | Prueba de contaminación de parámetros HTTP | Hecho | Medio | 0 | 1 | 0 | 0 | 0 |
| Pruebas de validación de datos | OTG-INPVAL-005 | Validación de entrada | Prueba de inyección SQL | Hecho | Alto | 1 | 0 | 0 | 0 | 0 |
| Pruebas de validación de datos | OTG-INPVAL-006 | - | Prueba de inyección LDAP | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Manejo de errores | OTG-ERR-001 | Manejo de errores | Análisis de códigos de error | Hecho | Ninguno | 0 | 0 | 0 | 1 | 0 |
| Manejo de errores | OTG-ERR-002 | - | Análisis de rastros de pila | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Criptografía | OTG-CRYPST-001 | Configuración | Pruebas de cifrados SSL / TSL débiles, protección insuficiente de la capa de transporte | Hecho | Alto | 1 | 0 | 0 | 0 | 0 |
| Criptografía | OTG-CRYPST-002 | - | Prueba de relleno de Oracle | N/A | n/a | 0 | 0 | 0 | 0 | 1 |
| Criptografía | OTG-CRYPST-003 | Criptográfico | Prueba de información confidencial enviada a través de canales no cifrados | Hecho | Medio | 0 | 1 | 0 | 0 | 0 |

Fuente: elaboración propia

La Figura 73, muestra la estadística de las pruebas realizadas, cada una de ellas con su respectivo nivel de riesgo, la cual, indica que existe un total de 57 pruebas realizadas, para el riesgo denominado Alto, se registran 10 pruebas, para el riesgo denominado medio, se registran 11 pruebas, para el riesgo denominado bajo, se registran 12 pruebas y para las pruebas que no determinan ningún tipo de riesgo existen 12.

**Figura 75.**

*Gráfico estadístico general de la métrica de riesgo por cada prueba realizada según OWASP v4.0*


Fuente: elaboración propia

En la Tabla 75, se muestra el riesgo por categoría de vulnerabilidad.

**Tabla 75.**

*Riesgo por Categoría de Vulnerabilidades*

| Categorías de Vulnerabilidades | Alto | Medio | Bajo | Ninguno | Total, Riesgos por Categoría |
|---|---|---|---|---|---|
| Abuso de API | 0 | 0 | 0 | 0 | 0 |
| Autenticación | 2 | 2 | 4 | 4 | 8 |
| Autorización | 1 | 1 | 1 | 2 | 3 |
| Disponibilidad | 0 | 0 | 0 | 0 | 0 |
| Permiso de código | 0 | 0 | 0 | 0 | 0 |
| Calidad del código | 0 | 0 | 0 | 1 | 0 |

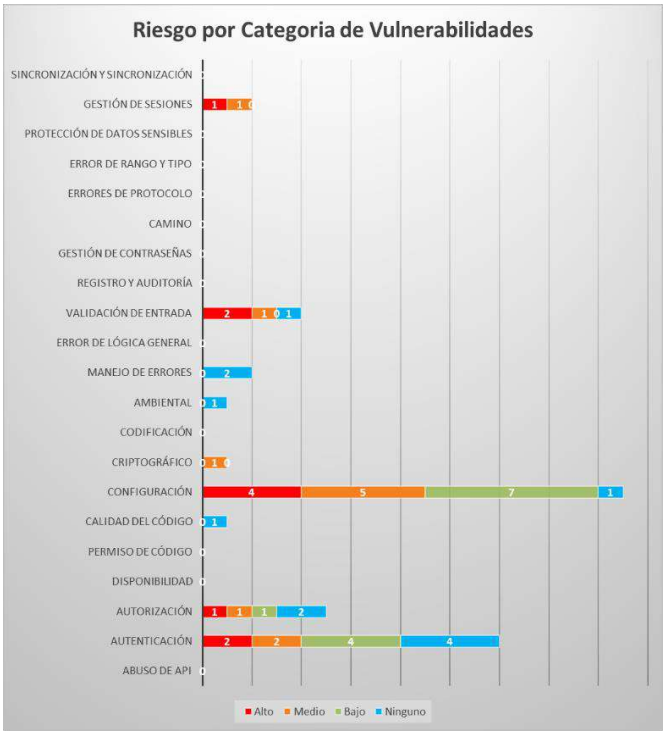
| | | | | | |
|---|---|---|---|---|---|
| Configuración | 4 | 5 | 7 | 1 | 16 |
| Criptográfico | 0 | 1 | 0 | 0 | 1 |
| Codificación | 0 | 0 | 0 | 0 | 0 |
| Ambiental | 0 | 0 | 0 | 1 | 0 |
| Manejo de errores | 0 | 0 | 0 | 2 | 0 |
| Error de lógica general | 0 | 0 | 0 | 0 | 0 |
| Validación de entrada | 2 | 1 | 0 | 1 | 3 |
| Registro y auditoría | 0 | 0 | 0 | 0 | 0 |
| Gestión de contraseñas | 0 | 0 | 0 | 0 | 0 |
| Camino | 0 | 0 | 0 | 0 | 0 |
| Errores de protocolo | 0 | 0 | 0 | 0 | 0 |
| Error de rango y tipo | 0 | 0 | 0 | 0 | 0 |
| Protección de datos sensibles | 0 | 0 | 0 | 0 | 0 |
| Gestión de sesiones | 1 | 1 | 0 | 0 | 2 |
| Sincronización y sincronización | 0 | 0 | 0 | 0 | 0 |
| Código móvil inseguro | 0 | 0 | 0 | 0 | 0 |
| Uso de API peligrosa | 0 | 0 | 0 | 0 | 0 |
| | | | | | |
| **Total de Riesgos** | **10** | **11** | **12** | **12** | **33** |

Fuente: elaboración propia

En la Figura 76, se muestra los resultados de las vulnerabilidades por categoría, la cual, indica que en la categoría configuración, se centran más los inconvenientes a tomar en cuenta, y tomar alguna medida para solucionar los inconvenientes se toma en cuenta las pruebas realizadas anteriormente.

**Figura 76.**

*Gráfico estadístico de Riesgo por Categoría de Vulnerabilidades*




Fuente: elaboración propia

Se toma en cuenta lo analizado anteriormente, por lo que se determina que existen vulnerabilidades a tomar en cuenta, las cuales se detallas a continuación:

- Falta Seguridad Strict Transport Security (HSTS) es un mecanismo de política de seguridad web.
- Vulnerabilidad Cross-Site Scripting (XSS) inyección de scripts maliciosos.
- Falta de un certificado SSL/TLS es un mecanismo de seguridad estándar para establecer un enlace cifrado entre un servidor y una aplicación web.
- Vulnerabilidad ante un ataque de Inyección SQL, ataque mediante comandos de base de datos.
- Administrar el manejo de Cookies que son fragmentos de datos que el navegador web almacena.
- Falta de una política de dominios cruzados de RIA aplicación de internet enriquecida.
- Configurar un cifrado de la Información para la aplicación web.
- Debilidad en el manejo de contraseñas

# CONCLUSIONES

- La fundamentación teórica y metodológica sobre métodos y técnicas usados ante amenazas en aplicaciones web, permite determinar que, La Guía de pruebas de OWASP v4.0, remite aproximadamente 90 pruebas, las cuales realizan un proceso de evaluación de la seguridad dentro de una aplicación web, para ello, se utilizan herramientas que ayudan a verificar si existen o no amenazas ante un posible ataque a un sistema web, dichas pruebas se han efectuado en su gran mayoría a la aplicación web de bibliotecas de la Universidad Técnica de Ambato, se toma en cuenta que existen parámetros que ayudan a determinan si cada prueba realizada se logra catalogar o no como una vulnerabilidad presente en la aplicación web, es por ello que al realizar dichas pruebas, se logró determinar que existen ciertas vulnerabilidades que alcanzan a ser consideradas una constante amenaza para la aplicación web, no cuentan con un nivel de seguridad aceptable.

- El Análisis de mecanismos válidos para resolver los problemas de vulnerabilidades en las aplicaciones web en la Universidad Técnica de Ambato que ofrece la metodología OWASP, permite concluir que, una vez concluidas las pruebas realizadas a la aplicación web de la institución mediante la metodología OWASP, se pude terminar que existe ciertas vulnerabilidades presentes en la aplicación web de bibliotecas, además, se pudo identificar falsos positivos dentro de la misma, la mayor cantidad de pruebas realizadas determinaron información importante como versiones de servidor web, lenguaje de programación, Frameworks y tipos de Software entre otras existen, además, vulnerabilidades a tomar en cuenta en algún trabajo futuro las cueles son e suma importancia para la pronta mitigación de las mismas, y así mantener segura la aplicación web ante alguna amenaza.

- En la presente investigación no se realizaron todas las pruebas, varias de ellas no se podían efectuar por cuestiones de programación, accesos restringidos y funcionalidad, así también, existen pruebas que no se pudieron efectuar, no existen ciertos mecanismos que exigen las pruebas para su correcta ejecución dentro de la aplicación web a ser analizada.

- La evaluación de las fases de la metodología OWASP relacionados a la seguridad de las aplicaciones web de la Universidad Técnica de Ambato, se concluye que, las pruebas

efectuadas a la aplicación web de bibliotecas de la Universidad Técnica de Ambato, se las ha efectuado mediante herramientas de Código Abierto, las cuales en su mayoría contienen funcionalidades limitadas con respecto a herramienta de pago, las cuales podrían tener funciones adicionales que ayuden a obtener información más precisa y clara en cuanto a la obtención de información se refiere, por ejemplo, Kali Linux es un sistema operativo que contiene herramientas de Código Abierto para realizar pruebas de penetración, dichas herramientas en su gran mayoría no cuenta con una interfaz gráfica, lo cual, dificulta el uso para obtener información que logra ser de gran utilidad.

# RECOMENDACIONES

- Se recomienda a la dirección de Tecnología de Información y Comunicación de la Universidad Técnica de Ambato, crear un plan de mitigación que incluya normas o políticas para la seguridad en aplicaciones web, para de esta manera mitigar o reducir los incidentes informáticos a los que alcanzan a ser víctimas los sistemas web de la Institución, y a su vez mantener segura cualquier tipo de información que sea confidencial, dentro la aplicación web de bibliotecas de la Institución, se logra detectar vulnerabilidades que suelen ser una amenaza para la integridad de la información presente dentro de esta.

- Antes de ejecutar las pruebas que propone la metodología OWASP en su Guía V4.0, se recomienda descartar aquellas que no se adaptan a la funcionalidad de una aplicación web, logran ser innecesarios, puesto que no reportaran ningún resultado, lo cual, implicaría pérdida de tiempo y dinero.

- Se recomienda realizar pruebas de penetración a todos los sistemas web de la institución, con la finalidad de reducir riesgos y a su vez identificar vulnerabilidades, se toma en cuenta que la metodología OWASP, también, cuenta con el TOP TEN de las principales vulnerabilidades que afectan a los sistema web, y que ejecutar estas pruebas a las diferentes aplicaciones, podrían ayudar a resolver problemas de seguridad en muchas de estas, principalmente a las que requieren mayor atención por la cantidad de información que podrían contener, y no causar una posible pérdida de tiempo o inclusive dinero que afecte la integridad del usuario o de la Institución.

- Se recomienda utilizar la guía resumida de las pruebas realizadas en la presente investigación las cuales se efectuaron como pauta la guía de pruebas de OWASP en la versión 4.0, las mismas que se encuentran detalladas en el Anexo 3, y las cuales consiguen ayudar a orientar de mejor manera el proceso para el realizar análisis de vulnerabilidades en las diferentes aplicaciones web dentro de la Universidad Técnica de Ambato.

## BIBLIOGRAFÍA

*Aprende que es una Deserialización Insegura*. (2019, julio 16). Binary Chaos - Hacking y Programación. https://hackingprofessional.github.io/Security/Aprende-que-es-Deserializacion-Insegura-OWASP-VII/

Campderrós Vilà, J. (2019). *Ataques y vulnerabilidades web*. http://diposit.ub.edu/dspace/handle/2445/143419

Chavarria Gonzalez, V. (2020). *Estudio de los ataques contra website. OWASP*. http://dspace.uib.es/xmlui/handle/11201/151259

*Cómo aplicar seguridad en el ciclo de vida del desarrollo de software*. (2020, febrero 14). Belatrix Software Development Blog. https://www.belatrixsf.com/blog/seguridad-desarrollo-software

El Mahjoubi, O. (2019). *Detección de vulnerabilidades y generación de alertas de seguridad para aplicaciones web*. http://openaccess.uoc.edu/webapps/o2/handle/10609/96087

Espíritu, C., & Alberto, D. (2018). Tecnología web con enfoque OWASP en la autenticación segura del registro en línea de menores del padrón nominado como aporte a la reducción de la brecha social de la primera infancia. *Universidad Nacional Federico Villarreal*. http://repositorio.unfv.edu.pe/handle/UNFV/2204

Global. Recuperado 22 de octubre de 2020. *SDLC and Development Methodologies*. de https://globaljournals.org/item/3918-sdlc-and-development-methodologies

INCIBE. (2020). *Proteccion de la informacion*. 33.

Intriago, A., & Karina, V. (2018). *PROPUESTA DE UNA METODOLOGÍA DE PRUEBAS DE PENETRACIÓN ORIENTADA A RIESGOS*. http://localhost:8080/xmlui/handle/123456789/2525

Li, J. (2020). Vulnerabilities mapping based on OWASP-SANS: A survey for static application security testing (SAST). *Annals of Emerging Technologies in Computing*, *4*(3), 1-8. Scopus. https://doi.org/10.33166/AETiC.2020.03.001

Marini, A., Miranda, E. A., Berón, M., Bustos, M. A., Riesco, D. E., & Rangel Henriques, P. (2019, abril). *Evaluación multicriterio sobre herramientas de análisis de seguridad en aplicaciones web*. XXI Workshop de Investigadores en Ciencias de la Computación (WICC 2019, Universidad Nacional de San Juan). http://sedici.unlp.edu.ar/handle/10915/77103

Molina, L., & Pilar, A. del. (2019). *Pentesting Web.* http://repository.unad.edu.co/handle/10596/25188

Niño Benitez, Y., Benitez, Y. N., & Martínez, N. S. (2018). Requisitos de Seguridad para aplicaciones web. *Revista Cubana de Ciencias Informáticas*, *12*(0), 205-221. https://rcci.uci.cu/?journal=rcci&page=article&op=view&path []=1787

*Open Information Systems Security Group*. Recuperado 27 de octubre de 2020, de https://www.oissg.org/

ostec. (2018, junio 7). Pentest: ¿qué es y cuáles son los principales tipos? *OSTEC Blog*. https://ostec.blog/es/seguridad-perimetral/pentest-concepto-tipos

Our Most Advanced Penetration Testing Distribution, Ever. Recuperado 7 de diciembre de 2020, de https://www.kali.org/

Overview of Information Security for New (and non-IT) Project Managers. Recuperado 22 de octubre de 2020, de https://www.linkedin.com/pulse/overview-information-security-new-non-it-project-managers-aakbar

OWASP Foundation | Open-Source Foundation for Application Security. Recuperado 27 de octubre de 2020, de https://owasp.org/

OWASP Risk Rating Methodology. Recuperado 3 de diciembre de 2020, de https://owasp.org/www-community/OWASP_Risk_Rating_Methodology

Pozos, T., & Inés, M. (2019). *Utilización de herramientas para pruebas de penetración en auditorias informáticas.* http://risisbi.uqroo.mx/handle/20.500.12249/2265

Pruebas de Penetracion. (2003, abril 20). *DragonJAR Seguridad Informática*. https://www.dragonjar.org/pruebas-de-penetracion.xhtml

¿Qué es el pentesting? Auditando la seguridad de tus sistemas. (2019, julio 4). INCIBE. https://www.incibe.es/protege-tu-empresa/blog/el-pentesting-auditando-seguridad-tus-sistemas

Quezada, A. E. C. Recuperado 10 de diciembre de 2020. *Pruebas de Penetración contra Aplicaciones Web*. 21.

Rani, S. B. A. S. U. (2017). A detailed study of Software Development Life Cycle (SDLC) Models. *International Journal of Engineering and Computer Science*, *6*(7), Article 7. http://103.53.42.157/index.php/ijecs/article/view/2830

RESEARCH. Recuperado 27 de octubre de 2020, de https://www.isecom.org/research.html

Reyes, M., & Javier, O. (2019). *Aspectos a tener en cuenta para el análisis de riesgos con base en las normas ISO/IEC 27001, ISO/IEC 27005 E ISO/ IEC 31000*. http://repository.unipiloto.edu.co/handle/20.500.12277/6350

Romaniz, S. C. (s. f.). Recuperado 10 de diciembre de 2020. *Seguridad de aplicaciones web: Vulnerabilidades en los controles de acceso*. 14.

Romero, V., & Yucenid, A. (2019). *Pentesting, ¿porque es importante para las empresas?* http://repository.unipiloto.edu.co/handle/20.500.12277/6286

Serna, O., & Andrés, C. (2019). *Amenazas, vulnerabilidades, factores de riesgo y defensa en profundidad en aplicaciones web*. http://repository.unipiloto.edu.co/handle/20.500.12277/4913

SGSI. Recuperado 20 de octubre de 2020, de https://www.iso27000.es/sgsi.html

Sharma, M. K. (2017). A study of SDLC to develop well engineered software. *International Journal of Advanced Research in Computer Science*, *8*(3), 520-523. https://doi.org/10.26483/ijarcs.v8i3.3045

Suarez, G., & Luis, J. (2020). *Importancia de la seguridad informática y ciberseguridad en el mundo actual*. http://repository.unipiloto.edu.co/handle/20.500.12277/8668

The ZAP Homepage. Recuperado 30 de noviembre de 2020, de /

Torres Hallo, M. (2020). *ARTÍCULO CIENTÍFICO: Modelo de gestión de riesgos de procesos de tecnologías de información bajo la norma ISO/IEC 27000 en empresas aéreas del Ecuador*. http://biblioteca.uteg.edu.ec/xmlui/handle/123456789/1176

Vasquez, C., & Arturo, C. (2019). *Pruebas de penetración e intrusión*. http://repository.unipiloto.edu.co/handle/20.500.12277/6273

Vulnerability Management Life Cycle | NPCR | CDC. (2019, marzo 12). https://www.cdc.gov/cancer/npcr/tools/security/vmlc.htm

Willberg, M. (2019). *Web application security testing with OWASP Top 10 framework* [Fi=AMK-opinnäytetyö|sv=YH-examensarbete|en=Bachelor's thesis|]. http://www.theseus.fi/handle/10024/170389

Zafra, G., & Luis, J. (2017). *Introducción al pentesting*. http://diposit.ub.edu/dspace/handle/2445/124085

Zapata, J. (2019). *Uso de tecnologías de pruebas de penetración para validación de seguridad de aplicaciones web basado en el top 10 de vulnerabilidades de OWASP*. http://repository.unad.edu.co/handle/10596/28466

# ANEXOS

## Anexo 1

## ENTREVISTA DIRECTOR DE LA DIRECCION DE TECNOLOGIA DE LA UNIVERSIDAD TECNICA DE AMBATO

1. Como se maneja la seguridad en las aplicaciones web de la institución.?

2. Cuáles son los procesos para mantener seguras las aplicaciones.?

3. Qué tipo de seguridad se aplica a las diferentes aplicaciones web.?

4. Conoce usted algún tipo de ataque informático a sitios o aplicaciones web.?

5. Conoce algún método o técnica para mitigar algún tipo de vulnerabilidad presente en una aplicación web.?

6. En la Dirección de Tecnología se maneja algún plan para evitar un ataque informático a las aplicaciones web.

7. Si las aplicaciones web se ven comprometidas por un ataque informático, se cuenta con los recursos informáticos y de personal para solucionar algún problema que logran generar dicho ataque.?

8. Se dispone en la Dirección de Tecnología los recursos informáticos para solucionar.?

9. Se realizan pruebas de seguridad a las aplicaciones web dentro de la Institución.?

10. ¿Cuál es o cuales son las aplicaciones cuyo uso sea más frecuente?

## Anexo 2

## Ficha de Observación 1

| | | |
|---|---|---|
| **Institución** | Universidad Técnica de Ambato | **Ficha N° 1** |
| **Dirección:** | Dirección de Tecnología de Información y Comunicación | |
| **Observador:** | Diego Leonardo Gamboa Safla | |
| **Fecha:** | | |

| **Ítems** | **Calificativos** | |
|---|---|---|
| | **Si** | **No** |
| ¿La infraestructura que procesa y almacena la información de las aplicaciones web está en área segura? | | |
| ¿Se cuenta con personal responsable de la seguridad de la Información dentro de la Institución? | | |
| ¿Conoce cuáles son las principales amenazas más usuales que se emplean a aplicaciones web? | | |
| ¿Se diseñan páginas de inicio de sesión lo suficientemente seguras para evitar el acceso a personas no autorizadas a aplicaciones web? | | |
| ¿Cree que es necesario redirigir al usuario a una nueva página después de un inicio de sesión? | | |
| ¿Existen en alguna política de seguridad en cuanto al manejo de la información dentro de la Institución? | | |

| | | |
|---|---|---|
| ¿Los usuarios de las aplicaciones web manejan algún tipo de seguridad para el manejo de contraseñas? | | |
| ¿Existe algún tipo de herramienta informática para realizar pruebas de seguridad a aplicaciones web? | | |
| ¿Se desarrolla software con todas las medidas de seguridad? | | |
| ¿Alguna aplicación web se ha visto afectada por algún tipo de ataque informático? | | |

**Ficha de Observación 2**

| | | |
|---|---|---|
| **Institución** | Universidad Técnica de Ambato | **Ficha N° 2** |
| **Dirección:** | Dirección de Tecnología de Información y Comunicación | |
| **Observador:** | Diego Leonardo Gamboa Safla | |
| **Fecha:** | | |

| **Ítems** | **Calificativos** | |
|---|---|---|
| | **Si** | **No** |
| ¿Conoce algún método o técnica para identificar vulnerabilidades en aplicaciones web? | | |
| ¿Conoce lo que implica un riesgo informático relacionado a una aplicación web? | | |
| ¿Existen algún proceso que indique que la integridad de la información está presente? | | |
| ¿Existen algún proceso que indique que la confidencialidad de la información está presente? | | |
| ¿Existen algún proceso que indique que la disponibilidad de la información está presente? | | |
| ¿Se toma alguna medida de protección para desarrollar una aplicación web segura? | | |
| ¿Las aplicaciones web se alojan en un sitio seguro hacer uso de estas? | | |

| | | |
|---|---|---|
| ¿Se tiene algún método o técnica para almacenar la información de las aplicaciones web en sitios seguros? | | |
| ¿Se realizan copias frecuentes de las aplicaciones web para evitar perdida de información de estas? | | |
| ¿Conoce los términos de riego, amenaza y vulnerabilidad? | | |

# Anexo 3

# GUÍA DE IMPLEMENTACIÓN RÁPIDA DEL MODELO OWASP

## 1.- Fundamentación

La presente guía de implementación rápida del modelo OWASP es un proceso que ayudara al desarrollo de diferentes pruebas a realizarse posteriormente en otros sistemas web de la Institución, esto se realiza mediante el proceso realizado a la aplicación web de Bibliotecas de la Universidad Técnica de Ambato en la cual, se ha efectuado alrededor de 90 pruebas para el análisis de vulnerabilidades dentro de dicha aplicación, se toma como referencia la guía de pruebas que ofrece la metodología OWASP en su versión 4.0.

## 2.- Objetivos

- Conocer las diferentes pruebas a realizarse a una determinada aplicación web para determinar si existe o no vulnerabilidades.
- Describir cada prueba para determinar el proceso a seguir en cada una de ellas.
- Aplicar las pruebas a realizarse a sistemas web que no dispongan de un nivel seguridad aceptable.

## 3.- Contenidos

### Recopilación de Información

Consiste en reunir la mayor cantidad de datos que sean de gran importancia para posteriormente ser organizados, analizados y sistematizados, con la finalidad de obtener información sensible dentro de un sistema web.

Para recabar información, es necesario tomar en cuenta lo que se desea buscar, para este caso lo que se busca obtener como principal información es:

- Diagramas de Red.
- Mensajes archivados y mensajes de correo electrónico.
- De los administradores y demás personal clave.
- Procedimientos de inicio de sesión y otros formatos de nombre de usuario.
- Nombres de usuario y contraseñas.
- Contenido de mensajes de error.
- Desarrollo, y versiones de una página web.
- Conocer el tipo de servidor web que es probado.
- Versión del servidor que se está ejecutando.
- Fuga de información de la ruta o rutas al directorio o carpeta de la aplicación web.

- Crear la lista de directorios que serían evitados por las arañas, robots o rastreadores.
- Herramientas DNS: nslookup, entre otros.
- Motores de Búsqueda: Google, Duck Duck Go, entre otros.
- DNS basados en la web: nmap, Nikto entre otros.
- Comentarios en el código fuente de páginas web.
- Metadatos en el código fuente de páginas web.
- Identificar dónde se utilizan peticiones GET y POST.
- Identificar todos los parámetros en las peticiones POST.
- Identificar todos los parámetros utilizados en una solicitud GET.
- Identificar todos los parámetros de la cadena de consulta enviados a través de una página web.
- Encabezados HTTP.
- Cookies.
- Códigos fuente HTML.
- Carpetas y documentos específicos.

Para obtener esta información estas pruebas, se utiliza herramientas de Código Abierto:

- Shodan.
- Google Docks.
- PunkSpider
- Httprint.
- Netcraft.
- Curl.
- Wget.
- OWASP: Zed Attack Proxy (ZAP).
- WASP: WebScarab.
- Burp Suite.
- Zed Attack Proxy (ZAP).
- WhatWeb.
- BlindElephant.
- Wappalyzer.

Para la toma o recopilación de la información, se recomienda utilizar herramientas adicionales a las mencionadas anteriormente, para obtener más información que ayude a determinar si existen o no vulnerabilidades adicionales.

Una vez recopilada la información, se procesa la información para determinar si existen o no vulnerabilidades que afecta a un determinado sistema web.

Además, se verificará que, si existe alguna vulnerabilidad detectada al momento de efectuar las pruebas al sistema web, logran comprometer la aplicación de la misma manera que una aplicación no segura alcanzan a comprometer al servidor.

## Pruebas de gestión de configuración e implementación

Comprender la configuración implementada del servidor que aloja la aplicación web es casi tan importante como las pruebas de seguridad de la aplicación en sí, por lo tanto, se tomara en consideración los siguientes aspectos.

- Los diferentes elementos que conforman la infraestructura serán determinados con el fin de comprender cómo interactúan con una aplicación web y cómo afectan a su seguridad.
- Todos los elementos de la infraestructura se revisan para asegurarse de que no contienen vulnerabilidades conocidas.
- Se hará una revisión de las herramientas administrativas usadas para dar mantenimiento a los diferentes elementos.
- Los sistemas de autenticación necesitan ser revisados para asegurarse que sirven a las necesidades de la aplicación y que no logran ser manipulados por los usuarios externos para obtener el acceso.
- Una lista de puertos definidos que se requieren para la aplicación recibirá mantenimiento y se guardará con un control de cambios.
- Sólo habilite módulos de servidor (extensiones ISAPI en IIS) que son necesarios para la aplicación
- Maneje los errores del servidor (40x o 50x) con páginas personalizadas en vez de usar las páginas genéricas del servidor web.
- Asegúrese de que el software del servidor se ejecuta con privilegios mínimos del sistema operativo.
- Asegurarse de que el servidor está configurado para manejar las sobrecargas adecuadamente y evitar ataques de denegación de servicio.
- Asegurarse de que el servidor ha sido calibrado correctamente en su rendimiento.
- Identificar las extensiones de archivo en uso dentro de las zonas conocidas de la aplicación (por ejemplo, jsp, aspx, html) y usar una lista básica de palabras con cada una de estas extensiones (o usar una lista más larga de extensiones comunes si los recursos lo permiten).
- Para cada archivo identificado a través de otras técnicas de enumeración, crear una lista personalizada de palabras, derivada de ese nombre. Obtener una lista de extensiones de archivo comunes (como ~, bak, txt, src, dev, old, inc, orig, copy, tmp, etc.) y utilizar cada extensión antes, después y en vez de la extensión del nombre del archivo actual.

Para obtener esta información estas pruebas, se utiliza herramientas de Código Abierto:

- Curl.
- Wget.
- Nessus.
- Nikto.

Las herramientas de desarrollo web suelen incluir instalaciones para identificar los enlaces rotos y los archivos no referenciados.

Los errores de configuración de un servidor web donde se aloja una determinada aplicación web, logra comprometer la información de esta, de la misma manera, una aplicación web no segura consigue llegar a comprometer el servidor.

Se recomienda realizar estas pruebas a todas las aplicaciones web que logran alojar un servidor, alcanza a darse el caso que no únicamente una aplicación web alcanzan a ver comprometida su información por falta de seguridad, sino todas las aplicaciones web.

**Pruebas de gestión de identidad**

En esta sección de pruebas, se verifica los diferentes roles de usuario y se realizan pruebas del proceso de registro. Recuperación de los detalles de la cuenta de usuario por registro o inicios de sesión fallidos, también, se prueba, así como la política de registro de nombre de usuario.

Para lo cual, se toma a consideración los siguientes aspectos:

- Manejo de Roles dentro de la aplicación web.
- Manejo de Usuarios dentro de la aplicación web.
- Verificar que los requisitos de identidad para registro de usuarios estén alineados con los requerimientos de seguridad y negocio.
- Validación del Proceso de Registro.
- Verificar qué cuentas logran aprovisionar otras cuentas y de qué tipo.
- Búsqueda de contraseñas y usuarios válidos.
- Analizar el código de error recibido en las páginas de inicio de sesión.
- Analizar URL y redireccionamientos de URL

Para obtener esta información estas pruebas, se utiliza herramientas de Código Abierto:

- Spidering.

El realizar estas pruebas en su gran mayoría son de forma manual, la validación de ciertos parámetros se verifica en la aplicación web publicada.

Validar aspectos como, el ingreso de datos informativos sea correctos, contraseñas seguras, usuarios validados, entre otros.

**Prueba de autenticación**

En esta sección de pruebas, se consiguen decir que Autenticación es establecer o confirmar algo como auténtico, es decir, que las afirmaciones hechas sobre alguna cosa son verdaderas. Autenticar un objeto significa confirmar su procedencia, mientras que si se habla de autenticar a un usuario o persona generalmente consiste en verificar su identidad. La autenticación depende de uno o más factores para su validación.

En seguridad informática, la autenticación es el proceso de tratar de verificar la identidad digital del remitente de una comunicación. Un ejemplo de este determinado proceso es el de inicio de sesión.

Probar un esquema de autenticación figura comprender cómo funciona el proceso de autenticación y usar esta información para evitar el mecanismo de autenticación.

Se tomará en cuenta los siguientes aspectos:

- Envío de datos con el método POST a través de HTTP.
- Envío de datos con el método POST a través de HTTPS.
- Envío de datos con el método POST a través de HTTPS en una página accesible a través de HTTP.
- Envío de datos con el método GET a través de HTTPS.
- Análisis de acceso a través de Fuerza Bruta mediante el uso de diccionario de datos.
- Determinar la resistencia de la aplicación web a la subversión del proceso de cambio de la cuenta que permite al usuario cambiar la contraseña de su cuenta.
- Determinar la resistencia de la función de restablecimiento de contraseñas para que no logren eludir o adivinar.

Es necesario, además, de estos aspectos a resolver responder ciertas preguntas que son de gran importancia para realizar pruebas de autenticación.

- ¿Cuál es el riesgo de forzado o adivinanza de contraseñas en la aplicación?
- ¿Basta un CAPTCHA para mitigar este riesgo?
- ¿Cómo se desbloquean las cuentas?

Para resolver estos aspectos, se las realizara mediante las siguientes herramientas de Código Abierto.

- OWASP Zap.

Se recomienda tener en cuanta, las opciones que presenta un determinado sitio web, las seguridades en cuanto al acceso al sistema se refieren, y la creación de usuarios y roles logran manejar para el acceso de la misma manera.

**Prueba de autorización**

La autorización es el concepto de permitir que el acceso a los recursos solo a aquellos usuarios a quienes se les otorga el permiso para usarlos. El probar la autorización significa comprender cómo funciona el proceso de autorización y utilizar dicha información para eludir algún mecanismo de autorización.

La autorización es un proceso que viene después de una autenticación exitosa, por lo que el evaluador verificará este punto después de que tenga credenciales válidas, asociadas con un conjunto bien definido de roles y privilegios. Durante este tipo de evaluación, se verificará si es posible omitir el esquema de autorización, encontrar una vulnerabilidad de recorrido de ruta o encontrar formas de escalar los privilegios asignados al evaluador.

Para resolver los aspectos anteriores, se detallan a continuación las siguientes pruebas:

- Pruebas para buscar el acceso a funciones administrativas.
- Pruebas para determinar el escalamiento de privilegios.
- Pruebas de la manipulación del rol/privilegio.

Para realizar estas pruebas, se utiliza las herramientas de Código Abierto.

- DotDotpwn.
- OWASP Zap.

Se recomienda para esta prueba verificar o comprobar que no sea posible escalar privilegios mediante la modificación de los valores de parámetros, además, el evaluador verificará que no es posible para un usuario modificar sus privilegios o roles dentro de la aplicación, de manera que podría permitir ataques de escalada de privilegios, así también, verificar si se implementó el esquema de autorización para que cada rol o privilegio obtenga acceso a funciones reservadas y recursos.

**Prueba de gestión de sesiones**

Uno de los componentes centrales de cualquier aplicación basada en web es el mecanismo mediante, el cual, controla y mantiene el estado de un usuario que interactúa con ella. Esto se conoce como administración de sesiones y se define como el conjunto de todos los controles que gobiernan la interacción de estado completo entre un usuario y la aplicación basada en web. Esto cubre ampliamente cualquier cosa, desde cómo se realiza la autenticación del usuario, hasta qué sucede cuando se desconecta.

HTTP es un protocolo sin estado, lo que significa que los servidores web responden a las solicitudes de los clientes sin vincularlos entre sí. Incluso la lógica de aplicación simple requiere que las múltiples solicitudes de un usuario se asocien entre sí a través de una "sesión".

Para realizar estas pruebas, se requiere analizar los siguientes criterios:

- ¿Están todas las directivas Set-Cookie etiquetadas como seguras?
- ¿Cualquier operación de cookie se lleva a cabo en un transporte no encriptado?
- ¿Logra la cookie ser forzada en un transporte no encriptado?
- Si es así, ¿cómo mantiene la aplicación la seguridad?
- ¿Hay cookies persistentes?
- ¿Qué tiempos para la caducidad se utilizan en las cookies persistentes y son estos razonables?
- ¿Son las cookies que se esperan sean transitorias configuradas como tal?
- ¿Qué ajustes HTTP/1.1 Cache-Control se utilizan para proteger las cookies?
- ¿Qué ajustes HTTP/1.0 Cache-Control se utilizan para proteger las cookies?

Mediante lo analizado anteriormente, lo que se busca encontrar en estas pruebas son:

- Colección de cookies.
- Analizar la sesión en la aplicación.

- Previsibilidad y aleatoriedad del identificador de sesión.
- Ingeniería inversa de cookies.
- Ataques de Fuerza Bruta.
- Probar las vulnerabilidades de proxys y caché.
- Probar Las vulnerabilidades De GET Y POST.
- Probar las vulnerabilidades de transporte.
- Probar el cierre de sesión desde el servidor.
- Verificar ataques (Cross-Site Request Forgery) CRSF.

Para realizar estas pruebas, se utiliza las herramientas de Código Abierto.

- Burp Suite.
- WebScarab.
- OWASP Zap.

Se recomienda verificar que en la aplicación web, se encuentre implementado un tiempo de cierre de sesión por inactividad. Este tiempo de cierre, define el período que una sesión se mantendrá activa en caso de que no haya actividad por parte del usuario. Cierre e invalide la sesión una vez excedida el período de inactividad definida desde la última petición HTTP recibida por la aplicación web para un determinado identificador de sesión.

Además, se tendrá en cuenta que, si una aplicación es vulnerable, el usuario está obviamente registrado cuando lee un mensaje que contiene un ataque CSRF, que logran apuntar a la aplicación de correo web y tenerla para realizar acciones como borrar mensajes, enviar mensajes que aparecen como enviados por el usuario, etc., y tomar este inconveniente para resolver de manera rápida y segura, y evitar así que un atacante tome el control de un sistema web.

**Pruebas de validación de datos**

La debilidad de seguridad de las aplicaciones web más común es la falla en validar adecuadamente la entrada proveniente del cliente o del entorno antes de usarla. Esta debilidad conduce a casi todas las principales vulnerabilidades en las aplicaciones web, como secuencias de comandos entre sitios, inyección SQL, inyección de intérprete, ataques locale / Unicode, ataques al sistema de archivos y desbordamientos de búfer.

Por lo que se analiza los siguientes aspectos:

- Probar ataques XSS.
- Utilizar una aplicación de acceso frontal e ingresar datos de entrada con caracteres especiales/no válidos.
- Analizar la respuesta de la aplicación.
- Identificar la presencia de controles de validación de información ingresada.
- Acceder al sistema de acceso restringido y verificar si los datos ingresados se almacenaron y cómo se almacenaron.

- Analizar el código fuente y entender cómo los datos almacenados, se procesan dentro de la aplicación.
- Pruebas de manipulación manual de verbos en HTTP.
- Elaborar solicitudes HTTP personalizadas (por ejemplo: OPTIONS, GET, HEAD, TRACE, CONNECT, entre otros).

Para realizar estas pruebas, se utiliza las herramientas de Código Abierto.

- Burp Suite.
- Curl.
- OWASP Zap.
- SqlMap.
- WebScarab.

Se recomienda que como el HTML estándar no es compatible con la petición de GET o POST, se tiene que crear solicitudes HTTP personalizadas para probar los otros métodos, además, se recomienda que los evaluadores comprueben cómo se procesa los ingresos del usuario por parte de la aplicación y cómo se almacena en el sistema de acceso restringido.

Así también, los ingresos almacenados por la aplicación normalmente, se utiliza en etiquetas HTML, además, se encuentran como parte del contenido de JavaScript, por otro lado, las aplicaciones web que permiten a los usuarios almacenar datos están potencialmente expuestas al ataque XSS.

Los ataques de reflexión del Cross-site Scripting, se previenen mientras la aplicación web desinfecta la entrada de datos; una aplicación web de firewall bloquea la entrada maliciosa, o mediante mecanismos integrados en navegadores web modernos. El evaluador probará las vulnerabilidades asume que los navegadores web no impedirán el ataque. Los navegadores consiguen estar desactualizados o con sus características de seguridad incorporadas deshabilitadas. Del mismo modo, los firewalls de la aplicación web no garantizan poder reconocer ataques nuevos y desconocidos.

**Manejo de errores**

El manejo de errores en las aplicaciones web, es muy indispensable, al controlar cada uno de ellos mediante algún proceso a prueba de fallos, hace que a aplicación sea más rápida y eficiente, es por ello por lo que las aplicaciones hoy en día necesitan de un control exhaustivo ante este tipo de problemas, que para un ciberdelincuente logran ser un paso importante para vulnerar una aplicación web.

Se pretende analizar los códigos más comunes (mensajes de error) y se enfoca en su importancia durante una evaluación de la vulnerabilidad.

Código de respuesta HTTP tales como 400 Bad Request, 405 Method Not Allowed, 501 Method Not Implemented, 408 Request Time-out and 505 HTTP Version Not Supported alcanzan a ser forzados por un atacante.

Para cumplir esta prueba, se realiza lo siguiente:

- Buscar errores en los servidores de aplicaciones.
- Manejo de errores en formularios web.
- Buscar errores de bases de datos.
- Prueba de errores: 404 Not Found.
- Prueba de errores: 400 Bad Request.
- Prueba de errores: 405 Method Not Allowed.
- Prueba de errores: 501 Method Not Implemented.
- Prueba de errores: 408 Request Time-out.
- Prueba de errores: 403 Forbidden.

Lo detallado anteriormente son los errores más comunes que se encuentran en sitios web, los cuales, el evaluador tendrá en cuenta al momento de realizar pruebas ante posibles fallos de un sistema web, determinar qué tipo de error emite ante un posible fallo, y en lo posible personalizar, para que dichos errores sean transparentes para el usuario.

Para realizar estas pruebas, se utiliza las herramientas de Código Abierto.

- Burp Suite.
- Zap Proxy.
- ErrorMint.

**Criptografía**

Los datos sensibles serán protegidos cuando se transmiten a través de la red. Dichos datos logran incluir credenciales y tarjetas de crédito del usuario. Como regla general, si los datos se protegerán cuando se almacenan, se protegerán también durante la transmisión.

El objetivo de la prueba es determinar si la aplicación web, contiene un certificado TLS/SSL (Transport Layer Security) / (Secure Sockets Layer), la cual, ayuda a mantener segura la aplicación web, además, se verifica si la información que se envía entre el usuario y la aplicación web, para que estas viajen de manera segura.

Además, se determinarán si los datos que se envía a través de la red son cifrados, si los datos que se envían se los realiza mediante HTTPS (Hypertext Transfer Protocol Secure), este mecanismo de protección no tendrá limitaciones o vulnerabilidades.

Por lo expuesto anteriormente, se analiza los siguientes aspectos:

- Verificar si existen banderas de seguridad para las Cookies de sesión.
- Verificar el uso de Seguridad estricta de transporte HTTP (HSTS).
- Analizar si existe la presencia tanto de HTTP como HTTPS, lo cual, se logra utilizar también para interceptar tráfico.
- Comprobar si existe instalado el certificado SSL – cliente y servidor.

Para realizar estas pruebas, se utiliza las herramientas de Código Abierto.

- Testssl.sh.
- Curl.

Por lo dicho anteriormente, se recomienda verificar que los datos sensibles serán protegidos al transmitirse a través de la red. Si los datos se transmiten a través de HTTPS o cifrados de alguna otra manera, el mecanismo de protección no tendrá limitaciones o vulnerabilidades, se evita que la información sea expuesta y ser tomada por un atacante.

**Conclusiones**

- Se logra utilizar el sistema operativo Kali Linux, el cual, contiene herramientas que ayudan a realizar pruebas de penetración, que tiene como finalidad buscar vulnerabilidad en aplicaciones web en la gran mayoría de sus herramientas.
- Se utiliza herramientas de paga que no se han utilizado dentro de la presente investigación, la cual, ayudaría a detectar cierta cantidad de información adicional que logra ser de gran ayuda para los procesos de evaluación a realizarse en un futuro.
- Se alcanza a utilizar otras herramientas que ayuden a encontrar información que muchas de estas suelen ser limitadas en cuanto al proceso de análisis se refiere.