Área: Ciencias de la Salud
Disciplina: Medicina
Tipo de artículo: Artículo de Revisión
ISSN: 2697-3316

DOI: 10.69825/cienec.v5i24.202

Vulnerabilidades en aplicaciones web utilizando la metodología de "proyecto abierto de seguridad de aplicaciones web"

Vulnerabilities in web applications using the "open project of web application security" methodology

Diego Leonardo Gamboa Safla[a,*]

a. Universidad Técnica de Ambato, Ambato, Ecuador; Email diego_gamboa1tr@hotmail.es

* Correspondencia: Diego Gamboa Safla, Email: diego_gamboa1tr@hotmail.es

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 efecto, un portal web ofrece una ventana que ciberdelincuentes logran utilizar como un medio para un ataque. En particular, 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.

Citation: Gamboa Safla, D.Vulnerabilidades en aplicaciones web utilizando la metodología de "proyectos abierto de seguridad de aplicaciones web".. *Revista Ciencia Ecuador* **2023**, *5*, 24. DOI: 10.69825/cienec.v5i24.202.

Received: 8/8/2023
Accepted: 25/10/2023
Published: 27/10/2023

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

Abstract: In the web applications at the Technical University of Ambato, information of vital importance is shown, on many occasions this information is static and in other dynamics. In effect, a web portal provides a window that cybercriminals can use as a means for an attack. In particular, it is important to carry out an analysis of software vulnerabilities that exist in the platforms that provide support at the Technical University of Ambato; To meet this objective, the OPEN WEB APPLICATION SECURITY PROJECT (OWASP) methodology is applied, which provides different approaches to vulnerability analysis that uses tools that help to perform penetration tests on web applications. In addition, it applies methods for resolution and mitigation of said vulnerabilities. This work is carried out on a certain web application of the Institution, during the second semester of the 2020 academic year, to demonstrate compliance with the objectives, different test guides for attacks on websites that use open source tools are used; With what is expected, that this solution contributes to computer security in the Institution's web application, thus obtaining a set of good practices in terms of security in web applications. The OWASP methodology provides different approaches for the analysis of vulnerabilities in web applications.

Keywords: Vulnerabilities, Open Web Application Security Project (OWASP), Computer Security.

1. Introducción

En los últimos años se ha observado intrusiones continuas a diferentes aplicaciones web : educativas, financieras o gubernamentales. Resaltan ataques a bases de datos y páginas web como 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.

Willberg 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 (1).

Por otro lado Chavarria V. manifiesta que típicamente a la hora de desarrollar una aplicación web no se dedica el tiempo y la importancia para incorporar las seguridades correspondientes, 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 (2). Destacan, entre los principales riesgos de seguridad a las aplicaciones web, las vulnerabilidades 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. (3)

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 y asegurar que la información 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 (3).

Los piratas informáticos suelen interesarse 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 incluirse, por ejemplo, las herramientas para iniciar sesión, proceso de compra en una página o funciones para administrar el contenido de una red social (4).

En el contexto ecuatoriano, Intriago A. et al. 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. (5). 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.

2. Materiales y Métodos

Para el desarrollo del presente proyecto, se aplica como modalidad de investigación la bibliográfica: libros técnicos, informes, artículos. 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.

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.

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.

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.

3. 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, En esta sección, se presenta los resultados obtenidos y previamente analizados, de las guías de la metodología OWASP v4.0. En la Tabla 1, 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 1. 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	Configura-ción	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	Autenti-cación	Proceso de registro de usuario de prueba	Hecho	Ninguno	0	0	0	1	0
Pruebas de gestión de identidad	OTG-IDENT-003	Autenti-cación	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 autenti-cación	OTG-AUTHN-001	Autenti-cación	Prueba de credenciales transportadas a través de un canal cifrado	Hecho	Alto	1	0	0	0	0
Prueba de autenti-cación	OTG-AUTHN-002	Autenti-cación	Prueba de credenciales predeterminadas	Hecho	Medio	0	1	0	0	0
Prueba de autenti-cación	OTG-AUTHN-003	Autenti-cación	Prueba de mecanismo de bloqueo débil	Hecho	Ninguno	0	0	0	1	0
Prueba de autenti-cación	OTG-AUTHN-004	Autenti-cación	Prueba para omitir el esquema de autenticación	Hecho	Medio	0	1	0	0	0
Prueba de autenti-cación	OTG-AUTHN-005	Autenti-cación	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 tabla 1, 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 1. ***Gráfico estadístico general de la métrica de riesgo por cada prueba realizada según OWASP v4.0***

Fuente: elaboración propia

En la Tabla 2, se muestra el riesgo por categoría de vulnerabilidad.

Tabla 2. *Riesgo por Categoría de Vulnerabilidades*

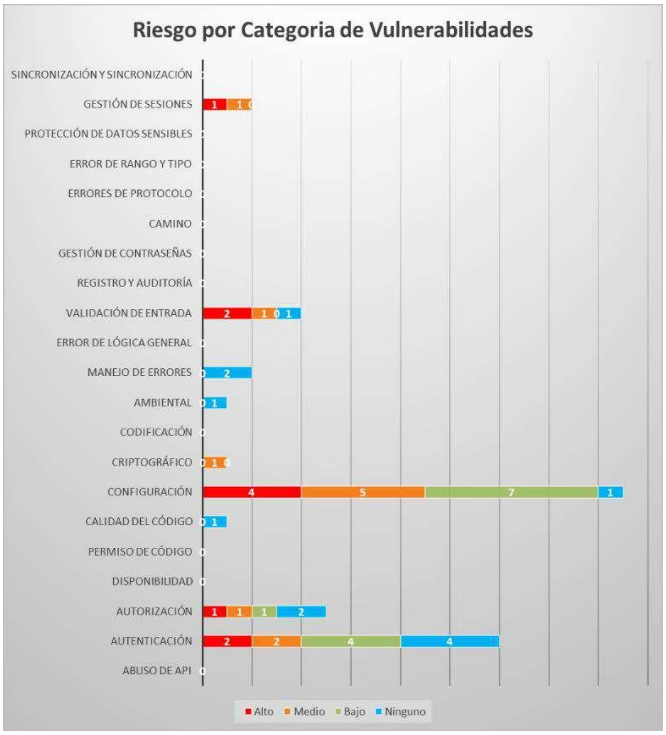
Categorías de Vulnerabilidades	Alto	Medio	Bajo	Ninguno	Total, Riesgos por Categoría
Abuso de API	0	0	0	0	0
Autenticación	2	2	4	4	8
Autorización	1	1	1	2	3

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 tabla 2, se muestra los resultados de las vulnerabilidades por categoría, la que, 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 2. Gráfico estadístico de Riesgo por Categoría de Vulnerabilidades



Fuente: elaboración propia

4. Discusión

Se observó las principales vulnerabilidades en una determinada aplicación web de la Universidad Técnica de Ambato. Según Serna et al. 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 (6) (7) (8).

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 (9) (10) (11) (12) (13).

Por otro lado Zapata, 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 (14).

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 (15)

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

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 (16) (17) (18)

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.

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.

Identificación de la responsabilidad y contribución de los autores: El autor declara haber contribuido en idea original, parte metodológica, redacción del borrador y redacción del artículo.

Revisión por pares El manuscrito fue revisado por pares ciegos y fue aprobado oportunamente por el Equipo Editorial de la revista CIENCIA ECUADOR.

Disponibilidad de datos y materiales Los datos que sustentan este manuscrito están disponibles bajo requisición al autor correspondiente.

Fuente de financiamiento Este estudio fue autofinanciado.

Conflicto de intereses Los autores declaran no tener conflictos en la publicación del presente manuscrito.

Bibliografía

1. Willberg, M. Web application security testing with OWASP Top 10 framework [Fi=AMK-opinnäytetyö|sv=YH-examensarbete|en=Bachelor's thesis|].2019 http://www.theseus.fi/handle/10024/170389.
2. Chavarria V. Estudio de los ataques contra website. OWASP.2020. http://dspace.uib.es/xmlui/handle/11201/151259.
3. OWASP. OWASP Top Ten.2020. https://owasp.org/www-project-top-ten/.
4. Jiménez J. Tipos de Ataques a aplicaciones web que debes conocer. RedesZone.2020. https://www.redeszone.net/tutoriales/seguridad/tipos-ataques-aplicaciones-web.
5. Intriago, A. Karina V. Propuesta de una metodología de pruebas de penetración.2018.
6. Serna O. Andrés C. Amenazas, vulnerabilidades, factores de riesgo y defensa en profundidad en aplicaciones web.2019. http://repository.unipiloto.edu.co/handle/20.500.12277/4913.
7. Marini A, Miranda E, Berón M, Bustos M , Riesco D, Rangel P. Evaluación multicriterio sobre herramientas de análisis de seguridad en aplicaciones web. XXI Workshop de Investigadores en Ciencias de la Computación.2019.
8. Molina L, Pilar A. Pentesting Web.2019. http://repository.unad.edu.co/handle/10596/25188.
9. Akamai. Como mejorar con akamai las prácticas de seguridad para mitigar los 10 principales ries-gos.2020.
10. Campderrós J. Ataques y vulnerabilidades web. 2019.
11. Sharma M. A study of SDLC to develop well engineered software. International Journal of Advanced Research in Computer Science.2017. 8(3), 520-523.
12. Suarez G, Luis J. Importancia de la seguridad informática y ciberseguridad en el mundo actual.2020.. .
13. Torres M. Modelo de gestión de riesgos de procesos de tecnologías de información bajo la norma ISO/IEC 27000 en empresas aéreas del Ecuador.2020.
14. Zapata J. Uso de tecnologías de pruebas de penetración para validación de seguridad de aplicaciones web basado en el top 10 de vulnerabilidades de OWASP.2019. http://repository.unad.edu.co/handle/10596/28466.
15. Espíritu C, Alberto D.. 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.2018. Universidad Nacional Federico Villarreal.
16. El Mahjoubi O. Detección de vulnerabilidades y generación de alertas de seguridad para aplicaciones web.2019. http://openaccess.uoc.edu/webapps/o2/handle/10609/96087.
17. Rani S. A detailed study of Software Development Life Cycle (SDLC) Models. International Journal of Engineering and Computer Science.2017. 6(7).
18. Reyes M, Javier O. 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. 2019.
19. Niño Y, Benitez Y, Martínez N. Requisitos de Seguridad para aplicaciones web. Revista Cubana de Ciencias Informáticas.2018 12(0), 205-221.