Artículo de Investigación. Revista Killkana Técnica. Vol. 2, No. 1, pp. 13-18, enero-abril, 2018.
ISSN 2528-8024. ISSN Elect. 2588-0888. Universidad Católica de Cuenca

Evaluación de seguridad de la información en las páginas web pertenecientes a los municipios de la provincia del Cañar

Information security assessment on web pages belonging to the municipalities of the province of Cañar

Cristhian Flores Urgilés[1]*, Beatriz Zhinin Aguayza[1], Alexandra Segovia Cantos[1], Mercedes Mayancela Zhinin[1], Jessica Marlene García[1]
Ingeniería en Sistemas, Universidad Católica de Cuenca extensión Cañar[1]
*chfloresu@ucacue.edu.ec

DOI: https://doi.org/10.26871/killkana_tecnica.v2i1.286

Resumen

La presente investigación se desarrolló con el fin de buscar el aseguramiento de la información de las páginas web pertenecientes a las municipalidades de la provincia del Cañar mediante un análisis de vulnerabilidades, buscando prevenir las amenazas que se puedan presentar y estas a la vez pongan en peligro la estabilidad de dichas páginas. Se realizó una investigación teórica sobre aseguramiento de aplicativos webs, estándares y buenas prácticas de seguridad de la información y las diferentes herramientas de escáner de vulnerabilidades de páginas web, para finalmente luego del análisis respectivo utilizar la herramienta Acunetix 11, además tomando como referencia el Top Ten de Owasp 2017. Como resultado se obtuvo un informe detallado de las vulnerabilidades encontradas durante el proceso de escaneo de las diferentes páginas municipales; para finalizar con las respectivas recomendaciones de seguridad para las amenazas más comunes.

Palabras clave: Owasp, Vulnerabilidad, informática, seguridad de información, amenaza informática, escaneo de vulnerabilidades.

Abstract

The present research was developed in order to seek the security of the information in the web pages of the municipality from the Cañar Province through a vulnerability analysis, seeking to prevent the threats that they can present and these can risk the stability of such web pages. First, a theoretical investigation about the security of web applications, standards, good information security practices, and the different vulnerability scan tools of web pages was carried out. Finally, after the respective analyses the Acunetix 11 tool was used, taking as reference the OWASP 2017 Top Ten security risks. As a result, a detailed report about the vulnerabilities that were found during the scanning process of the different municipal web pages was obtained, to conclude with the respective security recommendations for the most common threats.

Key words: OWASP, Informatics Vulnerability, information security, computer threats, vulnerability scanning.

I. Introducción

En la actualidad la mayoría de las empresas públicas y privadas utilizan sitios web y aplicativos móviles como herramientas fundamentales para informar sus actividades o para automatizar algunos de sus procesos administrativos y operativos; en vista de que los incidentes de seguridad en este caso involucrarían la confidencialidad, integridad o disponibilidad de todos los datos asociados al aplicativo web, así como también a los usuarios del mismo.

Consientes que desde el primer momento en que un servidor, computadora, se conecta a internet, se encuentra expuesto a diferentes tipos de amenazas informáticas, la mayoría derivada de vulnerabilidades propias tan simples como la versión de sistemas operativos u otros más complejos, en vista que los cambios tecnológicos ocurren rápidamente, lo que implica que a las actualizaciones de los diferentes programas, sistemas operativos y controladores de hardware se tengan que realizarse periódicamente, caso contrario se convertirán en vulnerabilidades que podrían ser explotadas.

En la actualidad se acostumbra a utilizar referentes o buenas prácticas de seguridad de aplicativos webs como una forma de analizar los riesgos más comunes a los que los aplicativos se encuentran expuestos.

Mediantes software que escanean riesgos de seguridad

de aplicativos webs se permiten obtener reportes, comparándolos con los estándares, que servirán para determinar los riesgos existentes, para así implementar salvaguardas que permitirán minimizarlo.

II. Materiales y Métodos

A. Selección de la herramienta

La herramienta a utilizar para el escaneo de vulnerabilidades de los aplicativos webs fue Acunetix 11. Debido a que dicha herramienta es capaz de realizar un escaneo a cualquier sitio web a través del protocolo HTTP/HTTPS. Cabe resaltar que esta herramienta está actualizada a la última versión del Top ten de Owasp que es la del 2017.

Otra de las razones del porque se seleccionó la herramienta Acunetix 11 es que, al momento de terminar el análisis de vulnerabilidades de las páginas web, emite un informe detallado de cada uno de los riegos informáticos tomando como referencia al Top Ten de Owasp 2017, encontrados en los aplicativos webs analizados.

B. Población y muestra

Para el análisis de las páginas se realizó una investigación del número de cantones que tiene la provincia de Cañar.

La provincia de Cañar está conformada por 7 cantones de las cuales 6 de ellas tienen página Web solamente el cantón Suscal no posee página web.

Luego de haber realizado el análisis a cada una de ellas se llegó a obtener el resultado que a continuación se detallara.

C. Metodología

- Como primer paso fue la identificación del dominio URL, de cada uno de los municipios que poseen sitios web.
- Luego de un tiempo de espera que es aproximadamente un día, tiempo que depende del contenido de cada una de las páginas.
- Una vez terminado el análisis emite un informe en donde se presenta todas las vulnerabilidades que tiene la página de acuerdo al Top Ten de Owasp 2017.
- En el siguiente paso Se procederá a la tabulación de los datos obtenidos para encontrar los riesgos informáticos más comunes.
- En base a los resultados del punto anterior se presentará una propuesta de prevención para los riesgos informáticos más comunes.

III. Desarrollo

A. Seguridad de la información

La seguridad informática consiste en proteger adecuadamente la información almacenada en un medio digital o no, en base a técnicas, estándares, medidas preventivas que nos ayuden a garantizar la protección y aseguramiento de esa información, el cual resulta ser el bien más valioso para las organizaciones que lo manejan. Permitiendo de esta manera garantizar la integridad, confidencialidad y disponibilidad de dicha información conocidos como los pilares fundamentales para asegurar la información, dichos conceptos en una aplicación web serian aplicados de la siguiente forma:

1) Confidencialidad es la propiedad que asegura que solo los que están autorizados tendrán acceso a la información.
2) La integridad es la propiedad que asegura la no alteración de la información.
3) La autenticación es la propiedad que hace referencia a la identificación. Es el nexo de unión entre la información y su emisor.
4) Disponibilidad La información estará disponible en cualquier momento solo a usuarios autorizados, evitando la divulgación.

Sin embargo, hoy en día no se puede ofrecer una protección absoluta de la información ya que cada vez existen nuevas amenazas, riesgos a las que se vuelven vulnerables los sistemas de información.

B. Seguridad en los aplicativos web

Las aplicaciones web vienen acompañadas de una variedad de vulnerabilidades de seguridad. Es por ello que las organizaciones se debe aplicar estándares, normas y reglas que ayuden a controlar a que la información y recursos estén protegidos.

1. Vulnerabilidades

Probabilidades que existen de que una amenaza se materialice contra un activo. No todos los activos son vulnerables a las mismas amenazas. Por ejemplo, los datos son vulnerables a la acción de los hackers, mientras que una instalación eléctrica es vulnerable a un cortocircuito. Al hacer el análisis de riesgos hay que tener en cuenta la vulnerabilidad de cada activo.[1]

2. Amenazas

Son agentes que poseen la cualidad de aprovechar una debilidad o vulnerabilidad para llevar a cabo un acto que afecte a un sistema. Las amenazas son directamente proporcionales a las falencias que tenga un sistema, si este posee grandes debilidades en su modo de operar o diseño, las amenazas serán altas y tendrán un gran impacto en el mismo.[1]

Amenazas más comunes para aplicativos webs

En la actualidad los aplicativos Web se encuentran en constante amenaza debido a que los ataques son cada vez más sofisticados. El Internet es el medio principal de ataque ya que, si no se ejecuta las medidas necesarias de seguridad, la información se vuelve vulnerable.

Entre los más comunes se puede citar los siguientes:

3. Injeccion

Las fallas de inyección, como SQL, NoSQL, OS e inyección LDAP, ocurren cuando los datos que no son de confianza se envían a un intérprete como parte de un comando o consulta. Los datos hostiles del atacante pueden engañar al intérprete para que ejecute comandos no deseados o acceda a datos sin la debida autorización.

4. Cross site scripting

Cross-site Scripting (XSS) se refiere al ataque de inyección de código del lado del cliente en el que un atacante puede ejecutar scripts maliciosos en un entorno legítimo. Sitio web o aplicación web. XSS ocurre cuando una aplicación web utiliza una entrada de usuario no validada o no codificada dentro de la salida genera

5. Broken Access control

Las restricciones sobre lo que los usuarios autenticados pueden hacer a menudo no se aplican correctamente. Los atacantes pueden explotar estos defectos para acceder a funcionalidades y / o datos no autorizados, como acceder a cuentas de otros usuarios, ver archivos confidenciales, modificar datos de otros usuarios, cambiar derechos de acceso, etc.

6. Broken authentication

Las funciones de aplicación relacionadas con la autenticación y la administración de sesión a menudo se implementan incorrectamente, lo que permite a los atacantes comprometer contraseñas, claves o tokens de sesión, o explotar otras fallas de implementación para asumir.

7. Sensitive data exposure

Muchas aplicaciones web y API no protegen adecuadamente los datos confidenciales, como financieros, de salud. Los atacantes pueden robar o modificar dichos datos protegidos débilmente para realizar fraude con tarjetas de crédito, robo de identidad u otros delitos. Los datos confidenciales pueden verse comprometidos sin protección adicional, como el cifrado en reposo o en tránsito, y requieren precauciones especiales cuando se intercambian con el navegador.

C. Análisis de Riesgos

Se denomina riesgo a la posibilidad de que se materialice o una amenaza aprovechando una vulnerabilidad. No constituye riesgo una amenaza cuando no hay vulnerabilidad ni una vulnerabilidad cuando no existe amenaza para la misma.[2]

El análisis de riesgo consiste en la cuantificación de dicho riesgo en basa a la probabilidad de ocurrencia conjuntamente con el impacto que causaría al sistema la materialización de la amenaza.

D. Estándares y Buenas prácticas de Seguridad en aplicativos Web.

1. Owasp

Es un proyecto de código abierto destinada a la de seguridad en aplicaciones Web, esta comunidad está dedicada a habilitar a las organizaciones para desarrollar, comprar y mantener aplicaciones confiables.[3]

¿Qué es el Owasp top 10?

El objetivo del proyecto Top 10 es crear conciencia sobre la seguridad de las aplicaciones mediante la identificación de algunos de los riesgos más críticos que enfrentan las organizaciones. Algunos estándares, libros, herramientas y organizaciones hacen referencia al proyecto Top 10, que incluye MITRE, PCI DSS, DISA, FTC y muchos más.[4]

2. Wasc (Web Application Security Consortium)

Los miembros Web Application Segurity Consortium, han unido esfuerzos de cooperación para desarrollar y promover un estándar de organización y clasificación de amenazas de seguridad web. Lo cual provee de un lenguaje común a desarrolladores, fabricantes de software, personal de seguridad y auditores, para tratar aspectos afines con la seguridad.[3]

Facilita un entendimiento profundo sobre los riesgos que amenazan a las aplicaciones Web. Wasc publica constantemente una guía información útil.

3. Herramientas de escáner de vulnerabilidades en aplicativos webs.

Estas herramientas permiten hacer un análisis a las páginas web para detectar las vulnerabilidades y riesgos de seguridad a las que están expuestas. Los escáneres de vulnerabilidad pueden ser de pago o gratuitas.

4. Grabber

Es unas herramientas más de escaneo de vulnerabilidades de las aplicaciones web, realizando exploraciones en las que se puede detectar amenazas como: SQL injection, Ajax Testing, etc. Se utiliza para analizar sitios web de cualquier tamaño debido a su fácil manejo.

5. Vega

Se puede utilizar para encontrar inyección SQL, inyección de cabecera, listado de directorios, inyección cáscara, Cross site scripting, la inclusión de archivos y otras vulnerabilidades de las aplicaciones web. Esta herramienta también se puede ampliar mediante una potente API desarrollada en JavaScript.[5]

6. Wapití

Realiza escaneos de çaja negra"(no estudia el código fuente) de la aplicación web al rastrear las páginas web de la aplicación web desplegada, en busca de scripts y formularios donde puede inyectar datos.[6]

7. *W3af*

W3af es un Marco de auditoría y ataque de aplicaciones web. El objetivo del proyecto es crear un marco para ayudarlo a proteger sus aplicaciones web al encontrar y explotar todas las vulnerabilidades de las aplicaciones web.[7]

8. *WebScarab*

WebScarab es un marco para analizar aplicaciones que se comunican mediante los protocolos HTTP y HTTPS. Está escrito en Java y, por lo tanto, es portátil para muchas plataformas. WebScarab tiene varios modos de operación, implementados por una cantidad de complementos. En su uso más común, WebScarab funciona como un proxy de interceptación, lo que permite al operador revisar y modificar las solicitudes creadas por el navegador antes de enviarlas al servidor, y revisar y modificar las respuestas devueltas desde el servidor antes de que el navegador las reciba.[8]

9. *Sqlmap*

Esta herramienta permite encontrar automáticamente las vulnerabilidades de sql Injection en la base de datos, con la que se encuentre trabajando el sitio web.

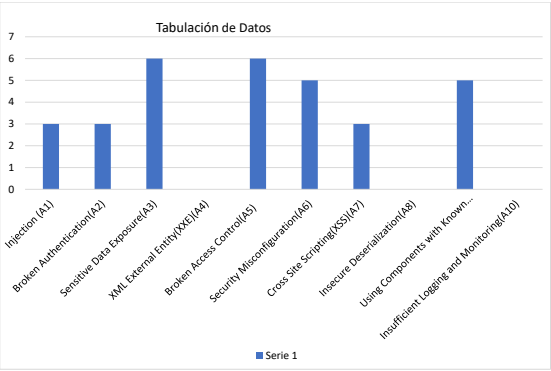
IV. Discusión

Luego de haber culminado el proceso de escaneo de vulnerabilidades utilizando la herramienta de Acunetix 11, en base al estándar Owasp Top Ten 2017 se obtuvo los siguientes resultados.

TABLA I. Vulnerabilidades encontradas

VULNERABILIDADES	TOTAL
Injection (A1)	3
Broken Authentication(A2)	3
Sensitive Data Exposure(A3)	6
XML External Entity(XXE)(A4)	0
Broken Access Control(A5)	6
Security Misconfiguration(A6)	5
Cross Site Scripting(XSS)(A7)	3
Insecure Deserialization(A8)	0
Using Components with Known Vulnerabilities(A9)	5
Insufficient Logging and Monitoring(A10)	0

Fig. 1. Ilustración del rango de vulnerabilidades



Fuente: Autor del proyecto

De acuerdo con los resultados que se muestran en la fig.1 se logró determinar las vulnerabilidades que son más perjudiciales en una página web, entre ellas están:

A. *Broken access control*

Esta vulnerabilidad es la que comúnmente se encuentra en las páginas web. Las restricciones sobre lo que los usuarios autenticados pueden hacer a menudo no se aplican correctamente, es decir que un usuario puede hacer clic en un enlace no deseado y dar paso a un atacante, para que este pueda acceder ya sea a cuentas de otros usuarios, ver y modificar Datos de usuarios, tomar control de la computadora. Por lo que el sitio web podría estar en riesgo de un ataque de clickjacking.

1. *Protección*

Para que una página este protegido ante esta vulnerabilidad se recomienda a los desarrolladores, realizar configuraciones tales como:

Frameguard: configura cabeceras X-Frame-Options en modo DENY ayuda a que no sea atacado por Clickjacking.

HidePoweredBy: desactiva cabeceras X-Powered-By evita que el atacante obtenga información.

XSS Filter: activa la cabecera X-XSS-Protection en modo "1; mode=block" para que el navegador puede detectar ataques XSS y los mismos puedan ser evitados.[9]

B. *Sensitive data exposure*

Esta vulnerabilidad es también una de las más comunes en páginas, permite a un atacante acceder a los datos confidenciales que no se encuentran protegidas, de esta manera pueden modificar dichos datos para realizar fraudes, robo de identidad.[3]

1. *Protección*

Una de las protecciones que se debe realizar para prevenir esta vulnerabilidad es clasificando datos procesados almacenados y transmitidos por un sistema. Aplicar controles según la clasificación.

No almacenar datos confidenciales innecesariamente hay que desecharlo una vez ya utilizado.

Existen vulnerabilidades que pocas páginas poseen, las cuales son:

C. *Security misconfiguration(a6)*

Las páginas web son expuestas a esta vulnerabilidad por su mala configuración de seguridad, estos detalles pueden ocasiones llegar a mostrar mensajes de error que contienen información confidencial, quedando expuestos a que una persona x pueda robar esa información y realizar chantajes a su conveniencia.

1. Protección

Para mejorar la seguridad de las páginas web y estar protegidos ante esta vulnerabilidad se sugiere, configurar de manera idéntica tanto el desarrollo, el control de calidad y los entornos de producción (con diferentes credenciales utilizadas en cada ambiente). Este proceso debe ser automatizado para minimizar el esfuerzo requerido para configurar un nuevo entorno seguro.

Eliminar o no instalar ninguna característica innecesaria, componentes, documentación y muestras. Eliminar los no usados dependencias y marcos.

D. Using components with known vulnerabilities

Provoca ataques que pueden facilitar la pérdida de datos graves o la toma del servidor.

1. Protección

Para esta vulnerabilidad se deben eliminar las dependencias no utilizadas, características innecesarias componentes archivos y documentación.

Se debe tener actualizado cada uno de los componentes que van a utilizar, en el lado del cliente y componentes del servidor

Se debe tener en cuenta que toda información personal no siempre se encontrase seguras, aunque existen páginas que se encuentran protegidos ante estas vulnerabilidades, según el análisis realizado existen muchas páginas que por su alta seguridad no permite su escaneo de vulnerabilidades.

V. Conclusiones

Se concluye que la herramienta más apropiada para evaluación de riesgos informáticos de aplicativos web es Acunetix 11, debido a sus excelentes características funcionales, una interfaz amigable, los informes se generan enbase a los estándares de buenas prácticas de seguridad, para el caso específico de nuestra investigación el Top Ten de Owasp 2017.

Del análisis realizado en los aplicativos webs de los municipios de la provincia de Cañar, tomando como referencia el Top Ten de Owasp 2017 los datos más relevantes son los siguientes:

El riesgo informático más común dentro de estas páginas web es Broken access control, que consiste en permitir a un usuario acceder a recursos y archivos para los que no debería tener permiso, provocando a los atacantes robar información. Para minimizar este riesgo informático se recomienda realizar configuraciones de cabecera X-Frame-Options en modo DENY que ayuda a que no sea atacado por Clickjacking, además desactivar la cabecera X-Powered-By, lo cual evita que el atacante obtenga información y por último activar la cabecera X-XSS-Protection en modo "1; mode=block" para que el navegador puede detectar ataques XSS y los mismos podrán ser evitados.

Además, otro de los riegos más frecuentes y que se debe tomar atención es Sensitive data exposure que consiste en que el atacante acceda a los datos confidenciales que no se encuentran protegidos, una de las recomendaciones que se puede plantear es clasificar datos procesados almacenados y transmitidos por un sistema de acuerdo al grado de confidencialidad de los mismos.

Para los riesgos menos frecuentes en el documento que precede se presenta una recomendación para minimizarlos, en vista de que cuando se realiza en forma planificada el aseguramiento de una página web es fundamental tomar en cuenta todos los aspectos relacionados, y sobre todo la prevención es la medida más importante en la seguridad de la información.

Referencias

[1] F. N. J. Solarte Solarte, E. R. ENRIQUEZ ROSERO, and M. d. C. Benavides Ruano, "Metodología de análisis y evaluación de riesgos aplicados a la seguridad informática y de información bajo la norma," 2015.

[2] A. López, *Seguridad informática*. Madrid: EDITEX, 2010.

[3] C. M. Flores Urgiles and C. H. Flores Urgiles, "ESPE.edu.ec," 2015.

[4] "OWASP," 2017.

[5] j. Mejía Viteri and G. J. Pérez Rueda, "utb.edu.ec," 2017.

[6] E. G. de Canales Glez, "Generación de reportes de vulnerabilidades y amenazas para aplicaciones web.," 2014.

[7] J. S. Monar Monar, "http://dspace.espoch.edu.ec," 2017.

[8] J. Orloff, "insureitsecurity.com," 2009.

[9] G. Gallardo Avilés, *Seguridad en base de datos y aplicativos web*. Madrid: IT Campus Academy, 2015.

***Recibido**: 3 de mayo de 2018*

***Aceptado**: 15 de junio de 2018*