Artículo **Revista de Cómputo Aplicado**
Marzo 2017 Vol.1 No.1 43-56

Implementación de mecanismos de seguridad en la aplicación web "BITA"

HERNÁNDEZ-CRUZ, Luz María*†, MEX-ÁLVAREZ, Diana Concepción, CAB-CHAN, José Ramón y MORA-CANUL, Ángel Leonardo

Universidad Autónoma De Campeche

Recibido Enero 4, 2017; Aceptado Marzo 7, 2017

Resumen

La presente investigación tiene como objetivo primordial mitigar las vulnerabilidades en la aplicación web "BITA" con la inclusión de mecanismos de seguridad. La investigación utiliza una metodología cualitativa, que inicia con el estudio del arte de la seguridad informática (SI) abarcando los conceptos básicos, la seguridad en sistemas de información (SSI), los diferentes tipos de ataques informáticos y los principales mecanismos de seguridad. En seguida, con el uso de la herramienta VEGA se identifican las vulnerabilidades en la aplicación y el equipo de desarrollo de software, utilizando la técnica Delphi, asigna un valor de prioridad para cada una de ellas. Por otra parte, se utiliza la técnica Grupo de discusión para elegir los mecanismos de seguridad que permitan mitigar dichas vulnerabilidades. Finalmente, se implementan los mecanismos dentro de la aplicación web "BITA". Este artículo aporta una visión actual de los diferentes mecanismos de seguridad destinados a conseguir un sistema de información seguro y confiable. Además contribuye a ostentar los beneficios de la inclusión de mecanismos de seguridad que permitan preservar la integridad, confidencialidad y disponibilidad dentro de la aplicación web "BITA".

Seguridad, aplicación web, vulnerabilidades, mecanismos de seguridad

Abstract

The primary objective of the following investigation is to mitigate vulnerabilities in the Web app "BITA" with the inclusion of security mechanisms. The investigation uses a qualitative methodology, which starts with the study of the art of the informatics security encompassing basic concepts, the security in information systems, the different types of computer attacks and the main security mechanisms. Right away, with the use of the VEGA tools the vulnerabilities are identified in the application and the software development team, using the Delphi technique, assigns a priority value for each one of them. On the other hand, we use the Discussion Group technique to choose the security mechanisms that allow mitigating such vulnerabilities. Finally, the mechanisms are implemented in the Web app "BITA". This article contributes a current vision of the different security mechanisms intended to achieve a secure and trustworthy information system. It also contributes to display the benefits of the inclusion of security mechanisms that allow preserve integrity, confidentiality and availability in the Web app "BITA".

Security, web application, vulnerabilities, security mechanisms

Citación: HERNÁNDEZ-CRUZ, Luz Marí†, MEX-ÁLVAREZ, Diana Concepción, CAB-CHAN, José Ramón y MORA-CANUL, Ángel Leonardo. Implementación de mecanismos de seguridad en la aplicación web "BITA". Revista de Cómputo Aplicado 2017, 1-1: 43-56

* Correspondencia al Autor (Correo Electrónico: lmhernan@uacam.mx)
† Investigador contribuyendo como primer autor.

Introducción

La Facultad de Odontología (FO) de la Universidad Autonóma de Campeche (UAC) brinda servicios odontológicos para contribuir con la sociedad Campechana, ofreciendo atención dental especializada en cinco clínicas dentro de su campus. La administración del proceso de atención al paciente y el control de expedientes clínicos odontológicos se realiza mediante un sistema web denominado "BITA".

"BITA" es una aplicación web diseñada por estudiantes de Ingeniería en Sistemas Computacionales (ISC) de la Facultad de Ingeniería (FI) que utiliza como lenguaje de programación JSP y como sistema gestor de base de datos MySQL. La implementación de la aplicación se realiza bajo un esquema de *proyecto de investigación interno, lo que ha permitido al personal directivo de la FO colaborar directamente en todo el ciclo de vida de desarrollo del software.*

Debido al tipo de información, de caracter crítico y confidencial, que involucra directamente la salud del paciente y el derecho a la privacidad de datos personales, surge la preocupación del personal directivo de la FO por analizar y mitigar los riesgos de seguridad dentro de aplicación web "BITA".

Esta investigación documenta la metodología y el análisis de vulnerabilidades de la aplicación web "BITA", así como los mecanismos de seguridad propuestos para mitigar dichas vulnerabilidades; cuyo valor agregado radica en servir de guía para su adaptación dentro de otras aplicaciones web.

El presente estudio se divide en las secciones que se mencionan en seguida:

a) Justificación: Apoya la importancia significativa de la investigación.

b) Estudio del arte: Fundamento teórico acerca de la Seguridad en los Sistemas de Información (SSI).

c) Metodología: Expone paso a paso el desarrollo del estudio de la investigación.

d) Resultados: Enfatiza los beneficios obtenidos.

Justificación

"BITA" es una aplicación web que manipula los expedientes clínicos de cada paciente atendido en las clínicas de la Facultad de Odontología de la UAC, con ello, es de suma importancia considerar el cumplimiento de la Ley Nº 25.326 "Protección de los Datos Personales" de Salud Pública en México, Capítulo II "Principios generales relativos a la protección de datos", que establece:

Artículo 9

1. El responsable o usuario del archivo de datos debe adoptar las medidas técnicas y organizativas que resulten necesarias para garantizar la seguridad y confidencialidad de los datos personales, de modo de evitar su adulteración, pérdida, consulta o tratamiento no autorizado, y que permitan detectar desviaciones, intencionales o no, de información, ya sea que los riesgos provengan de la acción humana o del medio técnico utilizado.

2. Queda prohibido registrar datos personales en archivos, registros o bancos *que no reúnan condiciones técnicas de integridad y seguridad*."

Por lo anterior, es clara la necesidad de "hablar" de seguridad en la aplicación web "BITA".

Según Baca (2016) se puede definir la seguridad informática o de la información de la siguiente manera:

"La seguridad informática es la disciplina que con base en políticas y normas internas y externas de la empresa, se encarga de proteger la integridad y privacidad de la información que se encuentra almacenada en un sistema informático, contra cualquier tipo de amenazas, minimizando los riesgos tanto físicos como lógicos a los que está expuesta."

Estudio del arte (Marco teórico)

Sin embargo, cuando contextualizamos la seguridad en una aplicación web, como es el caso de "BITA", no es lo suficientemente amplia la definición de Baca. "BITA" con solo el hecho de estar en Internet, ya hace potencialmente inseguros los datos que manipula.

Particularmente ha surgido una disciplina llamada seguridad de los sistemas de información que está en continua evolución.

Areitio (2008) menciona, en todo sistema de información, las principales actividades englobadas en el marco de la seguridad son (Véase Gráfico 1):

1. Seguridad de las operaciones o seguridad operacional. Se enfoca en la seguridad del entorno de las actividades y el mantenimiento de un entorno de trabajo seguro.

2. Seguridad de datos. Está relacionada con los datos y el mantenimieno de la seguridad durante su manipulación y procesado, tanto en sistemas fiables como las PC, estaciones de trabajo y servidores, como en las redes.

3. Seguridad de red. Implica la protección del hardware, del software y de los protocolos de red, incluyendo la información comunicada entre las redes.

4. Seguridad física. Se refiere a la protección del inmueble.

5. Seguridad del personal. Está relacionada con las personas, y sirve para determinar si son dignas de confianza y su concientización en materia de seguridad.

6. Seguridad administrativa. Son los aspectos de gestión de la seguridad dentro de la organización.

Figura 1 Actividades de Seguridad en Sistemas de Información (SSI)

Fuente: Areitio Bertolín, J., (2008). Seguridad de la Información. Redes, informática y sistemas de información. Madrid, España: Paraninfo.

Frecuentemente, los sistemas de información se hacen más vulnerables cuando se accesan por la Internet.

Según la asociación ACISSI (Auditoría, Consejo, Instalación y Seguridad de Sistemas de Información), existen dos ataques comunes para aplicaciones web: Inyección SQL y XSS (Cross-Site-Scripting). La inyección de SQL trata de aprovechar la protección inadecuada de ciertas peticiones SQL que usan parámetros. Estos parámetros han sido proporcionados por el usuario final. Típicamente, son datos que vienen de un formulario o de una URL.

Si estos párametros no se tratan adecuadamente, se podrán inyectar código malicioso y se permitirá accedar a los recursos de la base de datos. El ataque Cross-Site-Scripting (XSS), también intenta aprovecharse de un formulario o de los parámetros de una URL para inyectar datos de un sitio web, porque estos no se comprueban ni se protegen. Se puede provocar la ejecución de un script por parte del navegador web cuando el internauta visita la página. Mediante este script se pueden robar datos del usuario tales como sus cookies de autentificación, su sesión o redirigir el navegador a una página con la misma apariencia pero modificada por un hacker.

Por otro lado, también existen diversas normas y estándares que sirven de guía para administrar la seguridad en los sistemas de información, entre las que podemos mecionar, por su aceptación a nivel mundial, a ITIL e ISO.

ITIL (Biblioteca o Librería de Infraestructura de Tecnologías de Información) es un estándar internacional de mejores prácticas en la Gestión de Servicios Informáticos. "La clave en la implementación de ITIL como marco de mejores prácticas es proporcionar un servicio de alta calidad que le dé a la organización una distinción con respecto a sus competidores, el valor intangible que la organización ofrece a sus clientes. Comprender los objetivos del negocio del cliente y el rol que toma la organización que implementa ITIL para cumplir con las metas del negocio" (GUZMÁN, Agosto 2012). ISO (Organización Internacional de Normalización) es una organización internacional independiente, no gubernamental, con una membresía de 163 organismos nacionales de normalización. A través de sus miembros, reúne a expertos para compartir conocimientos y desarrollar estándares internacionales voluntarios, basados en el consenso y relevantes para el mercado, que apoyen la innovación y proporcionen soluciones a los retos globales.

La serie 27000 contiene las recomendaciones de mejores prácticas en Seguridad de la Información para desarrollar, implementar y mantener especificaciones para los Sistemas de Gestión de la Seguridad de la Información (SGSI). Entre ellas la ISO/IEC 27002 (Information Technology – Security Thechniques) Código de mejores prácticas para la gestión de seguridad de la información. (International Organization for Standardization, s.f.).

Esta investigación no incluye en su estudio el marco de referencia ITIL a causa de no ser, propiamente dicha, una guía para la gestión de seguridad de la información. El presente, únicamente se limita a considerar los objetivos clave de seguridad de la información recomendados por la Norma ISO.

Metodología

La metodología propuesta, pretende ser una alternativa para la implementación de seguridad en el desarrollo de aplicaciones web.

La metodología experimental se divide en tres fases principales a seguir, la primera, denominada *Análisis de Seguridad*, realiza un estudio de seguridad en la aplicación web "BITA" con el uso de la herramienta de software VEGA. La segunda, *Identificación de Vulnerabilidades*, permite comprender y analizar las vulnerabilidades detectadas y el tipo de daño o ataque informático que pueden causar. Por último, la tercera, *Mitigar Vulnerabilidades*, define el mecanismo de seguridad propuesto para mitigar las vulnerabilidades encontradas. (Véase Gráfico 2).

ISSN-2531-2952

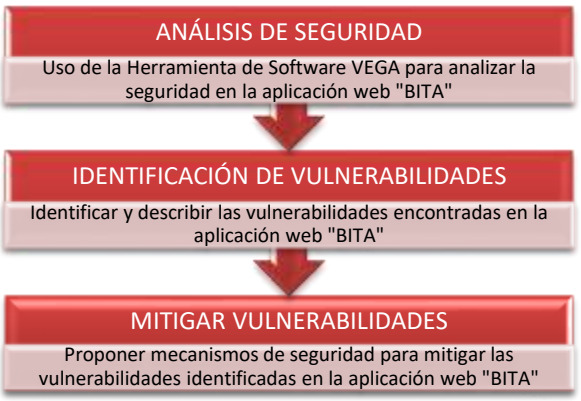

Figura 2 Metodología propuesta

Fuente: Fuente propia

Antes de iniciar con el desarrollo del estudio de esta investigación se establecen las dos variables principales a tratar: VI (variable independiente) representa las vulnerabilidades de seguridad encontradas y VD (variable dependiente) representa el mecanismo de seguridad propuesto para mitigar las vulnerabilidades encontradas.

Análisis de Seguridad

Primeramente, se detallan las principales características del proyecto "BITA" e inmediatamente cada una de las herramientas de software del entorno de prueba para llevar a cabo la fase de Análisis de Seguridad.

La aplicación web "BITA" fue desarrollada bajo el uso de la metodología de desarrollo de software SCRUM, que permite el desarrollo de software ágil. Este desarrollo se realiza de forma iterativa e incremental.

Según Paredes (2016) el equipo de Scrum consiste en tres diferentes roles:

- Product Owner: es el responsable de gestionar las necesidades que serán satisfechas por el proyecto y asegurar el valor del trabajo que el equipo lleva a cabo.
- Scrum Master: es el responsable de asegurar que el equipo siga las bases de Scrum.
- Scrum Team: El equipo está formado por los desarrolladores, que convertirán las necesidades del Product Owner en un conjunto de nuevas funcionalidades, modificaciones o incrementos del producto software final.

Para el proyecto "BITA", la Facultad de Odontología funge como propietario del producto (Product Owner), el Br. Ángel L. Mora Canul como líder del proyecto (Scrum Master) y el Equipo de desarrollo (Scrum Team) está integrado por Br. Julian Octavio Canul Pool, Br. Ramses Eduardo Martinez Santiago y Br. Jordy Manuel Can Uitz.
El lenguaje de programación empleado fue Java con tecnología JSP y MySQL como Sistema Gestor de Base de datos.

Actualmente MySQL ha tenido mucha aceptación a nivel mundial, porque tiene muy buen rendimiento, confiabilidad y facilidad de uso. Ésto le ha permitido colocarse como la principal opción para la creación de bases de datos de aplicaciones web (SÁNCHEZ Zindi, Marzo 2017).

El IDE (Entorno de Desarrollo Integrado) NetBeans v8.2 fue establecido para implementar la aplicación web "BITA". Según su página oficial, "Es un producto libre y gratuito (sin restricciones de uso). Es importante mencionar que NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios y una comunidad en constante crecimiento". Lo que implica que fácilmente se pueda dar mantenimiento a la aplicación web "BITA".

El programa GitHub, es una plataforma de desarrollo colaborativa de software que se emplea para la revisión, detección de problemas y comparación de cambios realizados en el código de la aplicación web "BITA". La Figura 3 muestra la interfaz principal del programa GitHub con el proyecto "BITA".

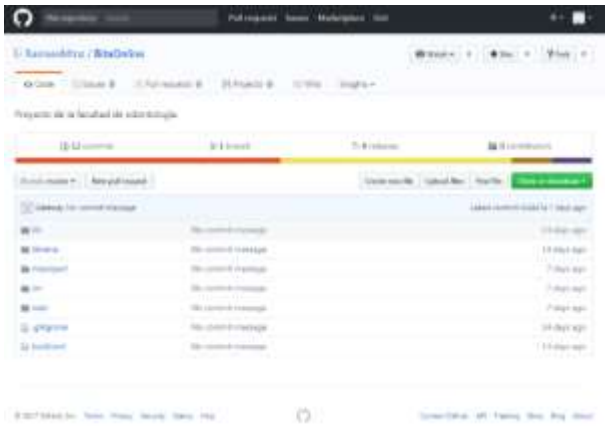

Figura 3 Entorno de prueba

Fuente: Fuente propia

El complemento NetBeans Connector v1.1 de Google Chrome admite la incorporación de Chrome con el software NetBeans. Permitiendo la sincronización de la aplicación web "BITA" con el navegador Chrome, esto ofrece una mayor flexibilidad al agregar, editar y eliminar código, con solo guardar los cambios y actualizar el navegador.

La herramienta de software VEGA v1.0 se ocupa para realizar la búsqueda de vulnerabilidades dentro de la aplicación web "BITA". VEGA permite encontrar vulnerabilidades como: Inyección SQL y Cross-Site Scripting, entre otras, recordando que estos dos tipos de ataques son básicos e indispensables en nuestro análisis.

Identificación de Vulnerabilidades

Se procede con el escaneo del módulo de acceso (Login) de la aplicación web "BITA" (Véase Anexo 1).

VEGA detecta un riesgo ALTO (High) indicando "Cleartext Password over HTTP". La Figura 4 muestra la información básica obtenida del análisis de la vulnerabilidad "Cleartext Password over HTTP".

Figura 4 Vulnerabilidad (VI_1) "Cleartext Password over HTTP"

Fuente: Fuente propia

VEGA detecta que el formulario de acceso contiene un campo de entrada de contraseña que envía a un destino inseguro (HTTP) los valores de contraseña. Esta vulnerabilidad podría conllevar a la propagación no autorizada de contraseñas.

Una segunda vulnerabilidad encontrada (VI_2) es "Sesión Cookie without secure flag". VEGA ha detectado que una cookie de sesión conocida puede haber sido establecida sin el indicador seguro, lo cual produce un riesgo ALTO (High). La Figura 3 muestra la información básica obtenida del análisis de la vulnerabilidad "Session Cookie Without Secure Flag".

Figura 4 Vulnerabilidad (VI_2) "Session Cookie Without Secure Flag"

Fuente: Fuente propia

VEGA detecta una tercera vulnerabilidad en la aplicación web "BITA", "Form Password Field with Autocomplete Enabled" (VI_3), en este caso, el nivel de riesgo es BAJO (Low). La Figura 4 muestra la información básica obtenida del análisis de la vulnerabilidad "Form Password Field with Autocomplete Enabled".

Figura 5 Vulnerabilidad (VI_3) "Form Password Field with Autocomplete Enabled"

El formulario de acceso a la aplicación web "BITA" tiene el atributo autocompletar habilitado. Esto puede dar lugar a que algunos navegadores almacenen los valores introducidos localmente por los usuarios, pudiendo ser recuperados por terceros.

Únicamente estas tres vulnerabilidades fueron detectadas por la herramienta de software VEGA al analizar cada módulo de la aplicación web "BITA" (Véase Anexo 1, 2 y 3). El equipo de desarrollo del proyecto emplea la técnica Delphi para asignar un valor de Prioridad a las vulnerabilidades encontradas, siendo el valor más crítico 1 (uno) y el menos crítico 3 (tres).
La Tabla 1 muestra los resultados obtenidos de las fases de Análisis de Seguridad e Identifición de Vulnerabilidades.

VI*n*	Vulnerabilidad	Nivel de Riesgo	Prioridad asignada
VI_1	Vulnerabilidad Cleartext Password	High	**1**
VI_2	Session Cookie Without Secure Flag	High	**2**
-	- Form Password Field with Autocomplete Enabled	Low	**3**

Tabla 9 Resultados obtenidos de la Fase de Análisis de Seguridad e Identificación de Vulnerabilidades

Mitigar Vulnerabilidades

Después de terminada la fase de Identificación de Vulnerabilidades, el equipo de desarrollo del proyecto "BITA" hace uso de la técnica Grupo de discusión para decidir los mecanismos de seguridad a implementar dentro de la aplicación, los cuales se describen a continuación.

Existen muchas maneras de obtener las contraseñas de acceso a una aplicación web, entre los métodos más comunes tenemos el ataque de fuerza bruta, la suplantación de identidad, y la interceptación. El acceso no autorizado a la aplicación web "BITA" es un tipo de vulnerabilidad alto y pone en riesgo toda la información que se almacena y manipula en ella, por tal motivo, el mecanismo de seguridad elegido para la administración de contraseñas seguras es mediante el cifrado. Se implementa SHA-256, una función hash de 64 dígitos hexadecimales que permite cifrar las contraseñas y que hace imposible su decodificación, cuando su empleo se realiza correctamente. Proviene de la familia de funciones hash de cifrado publicadas por el Instituto Nacional de Estándares y Tecnología (NIST) denominada SHA (Secure Hash Algorithm/Algoritmo de Hash Seguro). (Véase Anexo 4).

Un manejo inadecuado de Sesiones y Cookies, hace posible hackeos de información en una aplicación web. La clase java.servlet permite extender las capacidades de respuesta a los clientes al utilizar Java.

En la aplicación web "BITA" se hace uso del paquete *javax.servlet.http* para el manejo de sesiones mediante la interfaz HttpSession y los métodos que implementa. Asimismo, para el manejo de cookies se utiliza el constructor de la clase Cookie. Permitiendo un manejo fácil del uso de cookies y sesiones.

Desactivar la propiedad autocomplete del método POST en todos los formularios dentro de la aplicación web "BITA" es una medida de seguridad básica que permite eliminar vulnerabilidades (Véase Anexo 5 y 6).

La Tabla 2 muestra los resultados obtenidos al final de la Fase Mitigar Vulnerabilidades.

VI*n*	Vulnerabilidad	VD*n*	Mecanismo de Seguridad
VI_1	Vulnerabilidad Cleartext Password	VD_1	**Cifrado (SHA-256)**
VI_2	Session Cookie Whithout Secure Flag	VD_2	**Manejo de Servlet (cookies y sesiones)**
-	- Form Password Field with Autocomplete Enabled	VD_3	**Configurar propiedad Autocomplete (con deshabilitado)**

Tabla 10 Resultados obtenidos de la Fase Mitigar Vulnerabilidades

En este punto, se ha finalizado con la fase Mitigar Vulnerabilidades de la Metodología propuesta, consiguiendo eliminar las vulnerabilidades identificadas durante las dos primeras fases de la misma. Sin embargo, es inconcebible pensar que son las únicas vulnerabilidades a las que se expone la aplicación.

La norma ISO/IEC 27002 describe los objetivos que debe cumplir la seguridad, que son preservar la:

- Confidencialidad: significa que el acceso a la información se debe realizar únicamente por las personas autorizadas.

- Integridad: se refiere a la salvaguardia de la precisión de la información, ésta se encontrará completa y sin errores.

- Disponibilidad: las personas autorizadas a acceder a la información lo podrán hacer en el momento en que lo necesiten.

Como parte de enriquecer la presente investigación y aportar una arquitectura completa y funcional de seguridad básica dentro de una aplicación web, se añaden puntos críticos de la seguridad en la red y seguridad en el sistema gestor de base de datos implementados en la aplicación web "BITA" considerando los objetivos de la norma ISO/IEC 27002.

Seguridad en la red

La Internet, hace a una aplicación web vulnerable, las consideraciones mínimas para mitigar este riesgo son:

- Limitar el acceso a la red (Firewall). Todo el tráfico desde dentro hacia fuera, y viceversa de la red interna, debe pasar a través del Firewall. Sólo el tráfico autorizado, definido por la política local de seguridad, es permitido.

- Limitar el número de puntos de entrada (Puertos). Para evitar que un atacante obtenga información del sistema debe filtrar los puertos del servidor que no sean necesarios tener habilitados.

- Utilizar software legítimo.

- Utilizar herramientas de análisis de red para el escaneo de nuevas vulnerabilidades periódicamente.

Seguridad en el Sistema Gestor de Base de Datos

"Una base de datos relacional es una colección de información organizada en tablas para representar los datos y las relaciones entre ellos" (SÁNCHEZ Zindi, Marzo 2017). Para garantizar el correcto funcionamiento en la base de datos en la aplicación web "BITA" se ha considerado:

- Confidencialidad. Se crea un usuario de acceso "useradmin" para la manipulación de datos desde la aplicación web "BITA", restringuiendo a éste el acceso de las tablas. Utilizando sentencias preparadas dentro del código fuente de la aplicación. Igualmente se ha considerado una contraseña fuerte para el usuario (useradmin), aplicando la política longitud de ocho caracteres, contener por lo menos un carácter especial, una minúscula, una mayúscula y un dígito.

- Fiabilidad (Disponibilidad). Se ha diseñado un plan estratégico de copias de seguridad (backups) para poder recuperar la base de datos en caso de una falla, pérdida o catástrofe. El mecanismo de copias de seguridad no requiere una inversión adicional y su administración es sencilla como parte de la configuración propia dentro del sistema gestor de bases de datos.

- Trazabilidad. Se activa el registro binario, el registro de consultas y el registro de errores dentro del Sistema Gestor de Base de Datos para la supervisión y mantenimiento de la base de datos.

Resultados

El análisis de resultados se examina a partir de los controles de seguridad de acceso y operativa según la Norma ISO 27002 alineados a la metodología experimental propuesta. (ISO 27002, s.f.)

En la Fase de Análisis de Seguridad e Identificación de Vulnerabilidades se detectaron tres vulnerabilidades que se han tomado como referente para catalogar los controles de seguridad de Acceso en *atendido* (✓) y *no atendido* (×) dentro de la aplicación web "BITA". Del mismo modo, el análisis de la seguridad en la aplicación, en la red y en la base de datos son determinantes para catalogar el status de los controles en la Operativa. La Tabla 3 muestra el resumen del status *atendido* (✓) y *no atendido* (×) en la aplicación web "BITA" al inicio del estudio.

CONTROLES DE SEGURIDAD (ID)	Status	Referente asociado
ACCESO		
Control de acceso a sistemas y aplicaciones		
CA1. Restricción del acceso a la información	×	VI_1, VI_3
CA2. Procedimientos seguros de inicio de sesión	×	VI_2
CA3. Gestión de contraseñas de usuario	×	VI_1, VI_3
CA4. Control de acceso al código fuente de los programas	✓	SA
OPERATIVAS		
Protección contra código malicioso		
CO1. Controles contra el código malicioso	×	SA, SR, SBD
Copias de seguridad		
CO2. Copias de seguridad de la información	×	SBD
Gestión de la vulnerabilidad técnica		
CO3. Gestión de las vulnerabilidades técnicas	×	VI_1, VI_2, VI_3, SA, SR, SBD
CO4. Restricciones en la instalación de software	✓	SR
VI_1=Vulnerabilidad Cleartext Password, VI_2=Session Cookie Whithout Secure Flag, VI_3=Form Password Field with Autocomplete Enabled, SA = Seguridad en la Aplicación, SR = Seguridad en la red y SBD = Seguridad en la base de datos.		

Tabla 11 Controles de seguridad en la aplicación web "BITA" antes de implementar mecanismos de seguridad

Resumiendo, se tiene que de los 4 controles de seguridad de Acceso evaluados (CA1, CA2, CA3 y CA4), 3 no están atendidos dentro de la aplicación web "BITA", lo que equivale a un 75% de riesgo en el acceso a la aplicación. En el caso, de los 4 controles de seguridad de Operativa evaluados (CO1, CO2, CO3 y CO4) también 3 de ellos no están atendidos dentro de la aplicación web "BITA", lo que equivale a un 75% de riesgo en la Operativa de la aplicación.

Para medir el aprovisionamiento de seguridad después de aplicar los mecanismos de seguridad, se determina el nivel de riesgo asociado a cada control de seguridad evaluado (CA1, CA2, CA3, CA4, CO1, CO2, CO3 y CO4). Para ello, a las vulnerabilidades detectadas se les asigna el nivel de riesgo resultado del análisis con la herramienta de software VEGA, considerando nivel alto igual a 3, nivel medio igual a 2 y nivel bajo igual a 1. Por su parte, para la seguridad en la aplicación (SA) se ha desigando el nivel de riesgo medio, igual a 2; y para la seguridad en la red (SR) y en la base de datos (SBD) un nivel de riesgo alto, igual a 3. La Tabla 4 muestra el Nivel de Riesgo para cada control de seguridad evaluado.

CONTROL DE SEGURIDAD	Nivel de Riesgo (NR)	Nivel de Riesgo
CA1	**3**	**ALTO**
CA2	**3**	**ALTO**
CA3	**3**	**ALTO**
CA4	**-**	**N/A***
CO1	**3**	**ALTO**
CO2	**3**	**ALTO**
CO3	**3**	**ALTO**
CO4	**-**	**N/A****
Nivel de riesgo. ALTO= 3 (VI_1, VI_2, SR Y SBD), MEDIO =2 (SA) y BAJO=1 (VI_3). N/A (No aplica). *El acceso al código fuente de la aplicación web "BITA" sólo es parte del equipo de desarrollo, obedeciendo a las políticas propias de la Universidad. **La instalación de software es restringida por las políticas propias de la DGTI (Dirección General de Tecnologías de la Información) de la Universidad.		

Tabla 12 Nivel de Riesgo para los controles de seguridad evaluados en la aplicación web "BITA"

Si cada control de seguridad evaluado (CA1, CA2, CA3, CO1, CO2 y CO3) tiene un nivel de riesgo alto, tendríamos 6 x 3 = 18 como Valor Máximo de Riesgo (MaxR). En caso contrario, si se tiene un nivel de riesgo bajo, tendríamos 6 x 1 = 6 como Valor Mínimo de Riesgo (MinR).

Además, para cada control de seguridad se determinan los mecanismos o medidas de seguridad implementadas en la aplicación web "BITA", se asigna un nivel de Efectividad (alta = 3, media=2 y baja=1) y se realiza el cálculo del promedio de los datos de efectividad (Pe) usando la siguiente ecuación:

$$Pe = \sum \frac{Efectividad}{Total\ de\ Mecanismos} \tag{1}$$

Siguiendo el análisis de resultados al implementar los mecanismos de seguridad descritos en la metodología experimental presentada, se calcula el Riesgo residual (Rr) correspondiente a cada control de seguridad evaluado mediante la siguiente ecuación:

$$Rr = \frac{Nivel\ de\ riesgo}{\text{Promedio de Efectividad}} \tag{2}$$

La Tabla 5 muestra el análisis de resultados de cada control de seguridad y los mecanismos de seguridad implementados en la aplicación web "BITA".

ID (NR)	Medida o Mecanismo de Seguridad implementado	Efectividad
CA1 NR=3	Cifrado (SHA-256)	3
	GUI (Interfaz Gráfica de Usuario) (Configurar propiedad Autocomplete con deshabilitado)	2
PeCA1	Promedio de los datos de efectividad para CA1	2.5
	Riesgo residual CA1	1.2
CA2 NR=3	Manejo de Servlet (Cookies y Sesiones)	3
PeCA2	Promedio de los datos de efectividad para CA2	3
	Riesgo residual CA2	1
CA3 NR=3	Usuario (Nivel de acceso y Política de seguridad en la contraseña)	2
	Cifrado (SHA-256)	3
	GUI (Interfaz Gráfica de Usuario) (Configurar propiedad Autocomplete con deshabilitado)	2
PeCA3	Promedio de los datos de efectividad para CA3	2.33
	Riesgo residual CA3	1.28
CA4	N/A	-
PeCA4	Promedio de los datos de efectividad para CA4	-
	Riesgo residual CA4	-
CO1 NR=3	Uso y configuración de Contrafuegos (Firewall)	2
	Filtrado de Puertos	3
PeCO1	Promedio de los datos de efectividad para CO1	2.5
	Riesgo residual CO1	1.2
CO2 NR=3	Copias de Seguridad (Backups)	3
PeCO2	Promedio de los datos de efectividad para CO2	3
	Riesgo residual CO2	1
CO3 NR=3	Uso de Software Legítimo	2
	Uso de Software periódico para el Escaneo de la Red	2
	Configurar el Registro Binario, Registro de Consultas y Registro de Errores en el Sistema Gestor de Base de Datos.	3
PeCO3	Promedio de los datos de efectividad para CO3	2.33
	Riesgo residual CO3	1.28
CO4	N/A	-
PeCO4	Promedio de los datos de efectividad para CO4	-
Perfil de Riesgo (Total de Riesgo Residual)		7.7

Tabla 13 Análisis de resultados de los mecanismos de seguridad implantados en la aplicación web "BITA" de acuerdo a los controles de seguridad de Acceso y Operativa de la norma ISO 27002

Al obtener un *Perfil de Riesgo = 7.7* y compararlo con el *MaxR = 16* y el *MinR = 6*, podemos decir que se ha conseguido una arquitectura de seguridad sólida en la aplicación web "BITA". La Figura 5 muestra la Arquitectura de Seguridad implantada.

Figura 6 Arquitectura de Seguridad en la Aplicación Web "BITA"

Para finalizar, la Tabla 6 sintetiza los mecanimos de seguridad implantados en la aplicación web "BITA" como resultado del seguimiento al estudio realizado Para finalizar, la Tabla 6 sintetiza los mecanimos de seguridad implantados en la aplicación web "BITA" como resultado del seguimiento al estudio realizado.

Objetivo de Seguridad	Mecanismo de Seguridad
Confidencialidad	Cifrado (SHA-256)
Integridad	Manejo de Servlet (Cookies y Sesiones)
Confidencialidad	GUI (Interfaz Gráfica de Usuario) (Configurar propiedad Autocomplete con deshabilitado)
Confidencialidad	Contrafuegos (Firewall)
	Filtrado de Puertos
Integridad	Uso de Software Legítimo
Disponibilidad	Uso de Software periódico para el Escaneo de la Red
Confidencialidad	Usuario (Nivel de acceso y Política de seguridad en la contraseña)
Integridad	Backups
Trazabilidad	Configurar el Registro Binario, Registro de Consultas y Registro de Errores en el Sistema Gestor de Base de Datos.

Tabla 14 Mecanismos de Seguridad implementados en la aplicación web "BITA"

Anexos

Figura 6 Login de la Aplicación web "BITA"

Figura 7 Pantalla principal del módulo de Coordinación donde se encontraron vulnerabilidades

Figura 8 Pantalla principal del módulo de Encargada de clínica donde se encontraron vulnerabilidades

```
String password="secret";
    MessageDigest sha256=MessageDigest.getInstance("SHA-256");
    sha256.update(Password.getBytes("UTF-8"));
    byte[] digest = sha256.digest();
    StringBuffer sb = new StringBuffer();
    for(int i=0;i&lt;digest.length;i++){
        sb.append(String.format("%02x", digest[i]));
}
    String hash=sb.toString();
}
```

Figura 9 Encriptación SHA-256

```
<form method = "post" action = "/form" autocomplete = "off">
[...]
</form>
```

Figura 10 Campo de contraseña con la propiedad Autoacompletar desactivada

```
<form method = "post" action = "/form" autocomplete = "on">
[...]
</form>
```

Figura 11 Campo de contraseña con la propiedad Autocompletar habilitada

Agradecimiento

Se extiende un agradecimiento a M. en C. Juan Ricardo Oliva Luna, Director de la Facultad de Odontología, al M.C.C. Guadalupe Manuel Estrada Segovia, Director de la Facultad de Ingeniería, adscripciones de la Universidad Autónoma de Campeche, por el apoyo brindado en la realización del proyecto de desarrollo de la aplicación web "BITA", y por su dedicación, entrega y compromiso como parte del equipo de desarrollo del proyecto, a los estudiantes Br. Jordy Manuel Can Uitz, Br. Ramses Eduardo Martinez Santiago y Br. Julian Octavio Canul Pool.

Conclusiones

El presente artículo ha servido para el análisis de la aplicación web "BITA", un caso de estudio para ilustrar el uso básico de seguridad en una aplicación web.

Las oportunidades de considerar mecanismos de seguridad en Sistemas de Información (SI) específicamente en Aplicaciones Web es innumerable, ya que la creciente aparición de diferentes tipos de ataques y delitos informáticos hace que existan nuevas vulnerabilidades.

No cabe duda que se necesita concientizar y crear una cultura de incluir mecanismos de seguridad en los sistemas de información para garantizar, en la medida de lo posible, la seguridad en el uso de los mismos.

En un futuro cercano, las fortalezas de los sistemas de información radicarán en la seguridad que ofrezcan. Las Tecnologías de Información y Comunicación son el medio, pero los Sistemas de Información son el objetivo.

Queda un amplio campo para posteriores investigaciones, que consideren diferentes tipos de ataques y mecanismos de seguridad, incluso usando diferentes tecnologías de información y comunicación no incluidas en el alcance del presente.

Referencias

Areitio Bartolín, J. (2008). *Seguridad de la información. Redes, informática y sistemas de información.* Madrid : Parainfo.

Baca Urbina, G. (2016). *Introduccion a la Seguridad informática.* Grupo Editorial PATRIA.

Blogging googling. (11 de Abril de 2012). Obtenido de https://cirovladimir.wordpress.com/2012/04/11/java-obtener-el-hash-sha-256-de-una-cadena/

Burnett, M., & Foster, J. C. (2004). *Hacking the Code: ASP.NET Web Application Security.* Syngress Publishintg.

Castillo, L. (20 de Marzo de 2017). *Conociendo Github*. Obtenido de http://conociendogithub.readthedocs.io/en/latest/data/introduccion/

Didglee. (24 de Septiembre de 2016). *Mozilla Developer Network*. Obtenido de https://developer.mozilla.org/en-US/docs/Web/Security/Securing_your_site/Turning_off_form_autocompletion

GUZMÁN, Á. (Agosto 2012). ITIL v3 - Gestión de Servicios de TI. *Revista ECORFAN - México CÓMPUTO*, 801-806.

International Organization for Standardization. (s.f.). *International Organization for Standardization*. Obtenido de https://www.iso.org/

ISO 27002. (s.f.). *ISO 27002.es*. Obtenido de http://www.iso27000.es/iso27002.html

Joel Murach, A. S. (2008). *Murach's Java Servlets and JSP (2nd Edition).* Mike Murach & Associates, Inc.

Luz, S. D. (9 de Noviembre de 2010). *Redes Zone*. Obtenido de https://www.redeszone.net/2010/11/09/criptografia-algoritmos-de-autenticacion-hash/?utm_source=related_posts&utm_medium=manual

Marion AGÉ, F. E. (2015). *Seguridad informática Hacking Ético.* ENI.

Merino, F. G. (27 de Octubre de 2013). *Informatica y Comunicacion* . Obtenido de http://informatica.iesvalledeljerteplasencia.es/wordpress/plugin-netbeans-connector-1-1-de-google-chrome/

Mitnick, K. (2005). *The Art of Intrusión.* John Wiley & Sons.

Netbeans. (s.f.). Obtenido de https://netbeans.org/features/index.html

Paredes Xochihua, M., Morales Zamora , V., López Muñós, J., & Pedraza Varela , A. (Marzo de 2016). Diseño de sistemas para la simulación de metodologias de desarrollo de software . *Revista de Sistemas Computacionales y TIC´s, 2*(3), 22-29.

SÁNCHEZ Zindi, V. J. (Marzo 2017). Módulo de administración para una plataforma educativa del Instituto Tecnológico de Nogales. *Revista de Sistemas Computacionales y TIC's*, 19-24.

Schneier, B. (2000.). *Secrets & Lies. Digital Security in a Networked World* . John Wiley & Sons.

Semana. (30 de Enero de 2014). Obtenido de http://www.semana.com/tecnologia/tips/articulo/recomendaciones-para-tener-contrasena-segura/373739-3

Subgraph. (s.f.). Obtenido de https://subgraph.com/vega/