PROPUESTA DE AUDITORIA A LAS APLICACIONES WEB DE LA EMPRESA C&M CONSULTORES APLICANDO HERRAMIENTAS DE SOFTWARE LIBRE

JOHAN LORENZO CONTRERAS FLOREZ

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA UNAD
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA
ESPECIALIZACIÓN EN SEGURIDAD INFORMATICA
BOGOTÁ
2017

PROPUESTA DE AUDITORIA A LAS APLICACIONES WEB DE LA EMPRESA C&M CONSULTORES APLICANDO HERRAMIENTAS DE SOFTWARE LIBRE

JOHAN LORENZO CONTRERAS FLOREZ

Trabajo de grado para optar el título de
Especialista en Seguridad Informática

Director:
ANIVAR CHAVES TORRES
Ingeniero de sistemas

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA UNAD
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA
ESPECIALIZACIÓN EN SEGURIDAD INFORMATICA
BOGOTÁ
2017

Nota de aceptación

_____

_____

_____

_____

Presidente del Jurado

_____

Jurado

_____

Jurado

_____

Bogotá, junio 14 de 2017

DEDICATORIA

Dedico este trabajo a mi amada esposa, por su apoyo y ánimo que me brinda día a día para alcanzar nuevas metas, tanto profesionales como personales.

A mi adorado hijo Mattias Contreras Camargo, a quien siempre cuidaré para verlo hecho persona capaz y que pueda valerse por sí mismo.

A mis padres y hermanos, quienes son mi guía desde mi infancia.

A mis compañeros de trabajo, a quienes agradezco el apoyo y regaños por aprovechar un poco de tiempo del trabajo para elaborar la tesis.

AGRADECIMIENTOS

Agradezco a Dios y la virgen que me dieron la fuerza y fe para creer lo que parecía imposible terminar. A mis padres por ser los principales promotores de mis sueños, gracias a ellos por confiar y creer en mí y en mis expectativas.

A mis hermanos, por ser parte de mi vida y representar la unidad familiar, por ser un ejemplo de desarrollo profesional a seguir, por llenar mi vida de alegrías y amor cuando más lo he necesitado.

A mi esposa y mi hijo por impulsarme a terminar este proyecto.

Al Ing. Anivar Chaves Torres por brindarme su apoyo incondicional en todo momento, por sus valiosas asesorías, paciencia y consejos que me permitieron alcanzar los objetivos de la tesis.

CONTENIDO

Pág.

LISTA DE FIGURAS

Pág.

LISTA DE ANEXOS

Pág.

INTRODUCCION

Para dar inicio a este universo de la seguridad de la información a través de la web, es necesario implementar procedimientos seguros cuando se desarrollan aplicaciones web para compartir y publicar información, puede haber muchas opiniones o pensamientos, pero uno de los más críticos de la seguridad del Internet, lo tienen las piezas que intervienen de forma directa con las masas de usuarios, los servidores web. Es por ello que el tema de seguridad se debe asimilar como un engranaje en el que intervienen muchas áreas y técnicas cuando se trata de proteger la integridad de la información y la privacidad de los datos de los usuarios.

Este proyecto contiene la normativa para el desarrollo de aplicaciones web seguras en C&M CONSULTORES, empresa dinámica con más de 15 años de experiencia en Colombia prestando servicios de asesoría y consultoría gerencial dirigida al sector público.

No se trata de diseñar y de implementar aplicaciones que solo den una buena imagen y gusto a la empresa, sino también aspectos de seguridad que disminuyan el riesgo de pérdida de información o de manipulación de datos.

En el siglo XXI, los sistemas informáticos se han determinado en herramientas eficaces para administrar uno de los recursos más importantes de una empresa: sus sistemas de información. La Informática hoy forma parte de la gestión integral de la empresa, y, por lo tanto, son sometidos a las normas y estándares generales de la organización.

Debido a la gran incidencia en la administración o gestión de todas las actividades asignadas en la empresa, se efectúan Auditorias de Sistemas, con el fin de asegurar la eficiencia de las organizaciones, así como la confianza y seguridad de sus aplicaciones Web.

El este proyecto se abordó la propuesta de auditoria a las aplicaciones Web desde el enfoque de dos áreas específicas.

En primer lugar, la Auditoria de Aplicaciones consiste en tratar de ayudar a planificar, preparar y realizar auditorías de aplicaciones Web, en cuanto al

grado de cumplimiento de los objetivos para los que las mismas fueron creadas. De esta manera y, en consecuencia, se apoyó el logro de los objetivos organizacionales de la manera más satisfactoria.

En segundo lugar, la auditoria de explotación debe asegurar el funcionamiento adecuado de sistemas informáticos y su actualización. La detección oportuna de las debilidades del sistema permite mejorarlos racionalizando los recursos.

1. EL PROBLEMA DE INVESTIGACIÓN

1.1 DESCRIPCIÓN DEL PROBLEMA

En la actualidad la Empresa C&M CONSULTORES ha experimentado alguna transformación en algunos aspectos de seguridad, tales como: Impedimento de dar respuesta a aclaraciones de todo tipo sobre la información almacenada, diseñadas en contenido y forma para dar cobijo a las necesidades más comunes verificadas, No generar informes que usen de ayuda para cualquier finalidad de interés en la organización, presentando la información apropiada, La conexión cada vez más generalizada de la empresa a entornos abiertos como Internet triplica los riesgos que amenazan la confidencialidad e integridad de la información de nuestros sistemas, Evento de fallo en cualquiera de los elementos que intervienen en el proceso informático.

Para cada una de ellas se estudió las posibles medidas tendientes a eliminar los riesgos que introducen o, cuando menos, comprimir la probabilidad de su materialización hasta niveles razonablemente asumibles, siempre teniendo en cuenta el costo de tales medidas.

Dichas medidas son fundamentalmente medidas de control interno que consisten en los procedimientos para verificar, evaluar y tratar de garantizar que "todo" funciona como se espera; de acuerdo con las políticas, directrices, normas y procedimientos establecidos en los diferentes ámbitos de responsabilidad.

Así mismo, la situación actual permite conocer que los sistemas informáticos son el activo más valioso y al mismo tiempo el más vulnerable.

"Durante largo tiempo las aplicaciones web han sido vulneradas, permitiendo que personal externo o no autorizado tenga acceso a datos confidenciales, causando pérdida de información"[1].

Desde los inicios de esta empresa, se ha venido posicionando como una organización de primera línea en consultoría de alto nivel en las áreas de educación, infraestructura, bienestar social y tecnologías de la información y las comunicaciones; prestando servicios de interventoría, auditoría, gerencia y evaluación de proyectos dirigidos a los sectores público y privado.

---

1 UFPS, Plan de estudios evaluación de la seguridad física y lógica de los sistemas [En línea] http://repositorio.ufpso.edu.co:8080/dspaceufpso/bitstream/123456789/498/1/25972.pdf [Citado el 24 de mayo de 2017]

La seguridad informática ha adquirido un valor importante, dada las diferentes condiciones y nuevas plataformas de información disponibles, situación que implica en la aparición de nuevas amenazas en los sistemas informáticos.

Generalmente no se invierte ni el capital humano ni económico necesario para prevenir el daño y/o pérdida de la información confidencial en la empresa, a raíz de ello han surgido muchos problemas relacionados con el uso de computadoras, amenazas que afectan negativamente tanto a individuos como a la empresa; el crecimiento del uso de computadores como la principal herramienta, así como la creación de la red global Internet ha provocado que cada vez más personas se las ingenien para lucrar, hacer daño o causar perjuicios.

"El acceso no autorizado a un sistema informático, consiste en acceder sin ningún tipo de permiso, a un sistema de información, con el fin de obtener una ganancia de carácter intelectual y/o económico por el desciframiento de los códigos de acceso o claves"[2].

1.2 FORMULACIÓN DEL PROBLEMA

¿Cómo realizar una auditoría a las aplicaciones web de la Empresa C&M CONSULTORES aplicando herramientas de Software libre?

1.3 OBJETIVOS

1.3.1 General

Formular una propuesta de auditoria a las aplicaciones web de la empresa C&M CONSULTORES utilizando herramientas de Software libre que permitan determinar la seguridad de la información.

---

[2] INFOJUR, El acceso no autorizado a sistemas informáticos [En línea] http://www.buscalegis.ufsc.br/revistas/files/anexos/2778-2772-1-PB.html [Citado el 23 de mayo de 2017

1.3.2 Específicos

- Reconocer los tipos de vulnerabilidades, amenazas y riesgos de las aplicaciones web.
- Analizar los procedimientos, pruebas y herramientas de Software libre utilizadas para auditar una aplicación Web.
- Estudiar las particularidades de la Empresa C&M CONSULTORES
- Documentar la propuesta metodológica para determinar la seguridad de las aplicaciones Web en la Empresa C&M CONSULTORES

1.4 JUSTIFICACIÓN

La empresa C&M CONSULTORES se beneficiará con este proyecto, porque en los resultados del análisis de las aplicaciones web, se encontrarán los riesgos y vulnerabilidades más significativos haciendo uso de software libre, donde se podrán hacer rectificaciones a futuro en las políticas se seguridad establecidas, así como, educar a los empleados en la seguridad de las aplicaciones web y como tenerlas en cuenta al desarrollarlas.

El objeto de estudio, es el análisis completo del estado actual y futuro posible de las redes locales de la empresa C&M CONSULTORES. A través de la propuesta de auditoría se brindará una guía de recomendaciones que si bien no ofrecen la solución total contribuirá a la identificación de debilidades en las redes locales de la empresa.

En este contexto, controles propios de la empresa, “para establecer y analizar los activos de información, han dejado de ser “algo que hacen los de seguridad” para transformarse día a día en una disciplina que adopta la organización, para lograr como parte de su gestión, que la información sea una ventaja clave y competitiva frente a su entorno de negocio”[3].

Una aplicación web segura, es un elemento diferenciador, generador de confianza y valor para la empresa C&M CONSULTORES, además de esto, se busca obtener una metodología que pueda auditar la aplicación web segura en C&M CONSULTORES, basada en software libre.

---

3 CANO Jeimy J., Ciberseguridad y ciberdefensa: Dos tendencias emergentes en un contexto global [En línea], http://www.acis.org.co/fileadmin/Revista_119/Editorial.pdf, [Citado el 02 de septiembre de 2011]

1.5 DELIMITACIÓN

La investigación abarca el estudio y aplicación de las tecnologías utilizadas en el desarrollo de los servicios de consultas relacionadas con la web.

Específicamente, se pretende formular una propuesta metodológica para determinar la seguridad de las aplicaciones web en la empresa C&M CONSULTORES.

Este informe de trabajo de grado busca obtener una metodología, que brinde grandes beneficios como:

- Para la empresa C&M CONSULTORES que tiene presencia en la web, una herramienta metodológica que le permitirá determinar que vulnerabilidades posee la aplicación web.
- Para el departamento de sistemas, una herramienta útil para su trabajo que lo mantendrá a la vanguardia de las nuevas exigencias del mercado.

2 MARCO DE REFERENCIA

2.1 ANTECEDENTES

Entre las investigaciones realizadas anteriormente, se han tomado los siguientes antecedentes debido a la semejanza del problema y los excelentes resultados obtenidos.

- **Baena, E. (2006),** elaboró un proyecto titulado: “Aplicación Web para el control de Documentos de la Dirección de Protección Civil Bolívar”. Donde su objetivo fue el de Coordinar con la participación de las demás dependencias del Departamento, el diseño, dirección e implementación del sistema general de información administrativa del sector Público Integrado.

- **Figueroa, N. (2007)**, elaboró un proyecto titulado: "Diseño de un Sistema Computarizado para el Proceso de Facturación de la Empresa Inversiones Belmon Parr, C.A del Estado Monagas". Propone un sistema que lleve el control de la facturación de forma más segura, con claves únicas de acceso para cada facturador aplicando e implementando las modernas técnicas de administración y control. Y así poder garantizarle a la empresa la tranquilidad y el rendimiento de las inversiones hechas en ella.

- **Acosta, D. (2011),** elaboró un proyecto titulado: “Aplicación Web para el control de los Bienes Nacionales del Instituto Universitario de Tecnología del Estado Bolívar”. Propuesta que permite alcanzar la eficiencia en el control de inventario del departamento, buscando tener control general sobre el inventario que refleje el área del almacén, como dice ACOSTA.[4]

2.1 MARCO TEÓRICO CONCEPTUAL

2.1.1 Aplicaciones WEB

Es una aplicación a la cual se tiene acceso a través de un navegador Web sobre una red, ya sea Internet o una Intranet. Es de agregar que las aplicaciones web son codificadas en un lenguaje soportado por un navegador (Mozilla Firefox, Google Chrome, Internet Explorer, entre otros).

---

4 ACOSTA D. Aplicación Web http://yarmesm-aplicacionweb.blogspot.com.co/2011/11/capitulo-ii_10.html, [Citado el 22 de mayo de 2017]

“Una Aplicación Web es un tipo especial de aplicación Cliente/Servidor, en el que el cliente (Navegador Web), el Servidor (Servidor Web) y el Protocolo (HTTP) canal de comunicación están estandarizados y no es creado por el programador de aplicaciones”[5].

Figura 1. Arquitectura cliente servidor

Fuente: http://proypnfi.foroactivo.net/search.forum?search_author=Admin&show_results=posts

“El Cliente, en este caso el navegador, gestiona las peticiones del usuario y la recepción de las páginas que provienen del servidor, igualmente, interpreta los documentos HTML y sus recursos.

El Servidor, es el programa residente que espera peticiones: demonio (Daemon) en Unix y Servicio en servidores de Microsoft. En el servidor se encuentran páginas estáticas, Scripts o programas que al ser invocados se ejecutan y dan como resultado una página HTML”[6].

En la actualidad existen diferentes lenguajes de programación para desarrollar en la web, estos han ido evolucionando de acuerdo a las tendencias y necesidades de las plataformas. En inicios las aplicaciones web creadas fueron realizadas mediante lenguajes estáticos, posteriormente con el desarrollo y el avance de nuevas tecnologías surgen nuevas necesidades que dio lugar al desarrollo de Lenguajes Dinámicos de Programación que utilizan Bases de Datos y permiten interactuar con los usuarios.

---

5 Ingeniería de Requerimientos, Arquitectura de Tres Capas [En línea], http://www.indudata.com/index/capacitaciones/ingenieria-de-requrimientos.html, [Citado el 5 de septiembre de 2011]

6 DAEMON uuniversidad de Alicante, Qué es una aplicación Web [En línea], http://rua.ua.es/dspace/bitstream/10045/4412/5/03c-AplicacionesWeb.pdf, [Citado el 5 de septiembre de 2011]

2.1.2 Seguridad informática

"El objetivo de la seguridad informática será mantener la integridad, disponibilidad, privacidad, control y autenticidad de la información manejada por computadora"[7].

Figura 2. Seguridad Informática

Fuente: http://www.softtron.net/002/index.php/seguridad

En otras palabras, la seguridad Informática no es otra cosa que la capacidad de guardar intacta y protegida la información en un sistema informático. Sin embargo, la seguridad Informática abarca mucho más que la protección de la información, pero sin duda es el tesoro más atractivo para los *hackers*, teniendo en cuenta que la información es la base económica de las empresas.

"La Seguridad Informática se basa en tres principios fundamentales"[8]:

- ✓ Confidencialidad
- ✓ Integridad
- ✓ Disponibilidad

La ***confidencialidad*** es la seguridad de que los datos no son vistos por personas ajenas a la organización y que no tienen permiso para ello. Así pues, para controlar la confidencialidad de los datos se requiere la verificación y autorización.

La ***Integridad*** se centra en que la información no sea manipulada, alterada o cambiada por el sistema que la almacena o por entes externos no autorizados. Para mantener la Integridad de la información entre quien envía y quien recibe, se emplean técnicas criptográficas de cifrado que aseguran que la información no es modificada.

---

7 SEGURIDAD INFORMÁTICA, http://www.newwebstar.com/ebooks/133193-los-diferentes-lenguajes-de-programaciun-parala.html, [Citado el 6 de septiembre de 2011]

8 ALDEGANI, Gustavo. Miguel. Seguridad Informática. MP EDICIONES. ARGENTINA. 1997. Pág. 22

La ***Disponibilidad***, es el grado en que la información está en el lugar, momento y forma en que es requerido por personal autorizado, es decir, un sistema seguro debe mantener la información disponible para los usuarios que la requieran. Un posible método de ataque a un sistema informático es la denegación de servicio, que es lo opuesto a la disponibilidad, esto significa que el usuario no puede obtener del sistema los recursos requeridos.

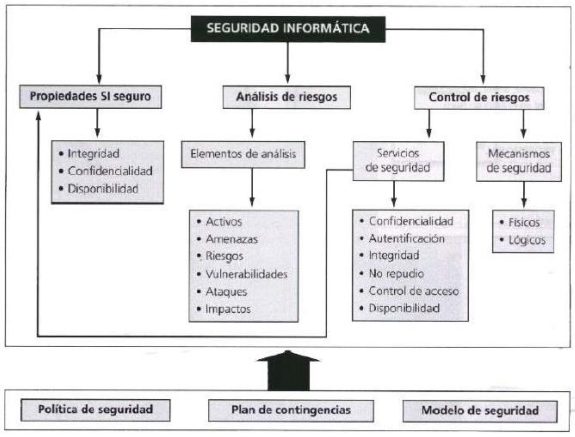
2.1.3 Modelos de seguridad

"[9]Un modelo de seguridad es la expresión formal de una política de seguridad y se utiliza como directriz para evaluar los sistemas de información. Los modelos de seguridad se pueden clasificar en tres grupos:

a) Matriz de acceso. Considera tres elementos básicos: el sujeto, objeto y tipo de acceso, es decir, un sujeto tiene o no permisos de acceso a un objeto del sistema. De esta manera se controla la integridad y confidencialidad de los datos.
b) Acceso basado en funciones de control (RBAC Role Access Base Control): en este caso el acceso no se define en función de quién es el sujeto sino de qué función tiene. Este modelo controla la confidencialidad y la integridad de los datos.
c) Multinivel. Se basa en la jerarquización de los datos, es decir, todos los datos son importantes, pero unos son más privados que otros".

El siguiente mapa conceptual resume a groso modo la Seguridad Informática:

Figura 3.. Resumen Seguridad Informática



Fuente: https://www.google.com.co/search?q=Resumen+Seguridad+Inform%C3%

[9] CRITERIOS, Control de accesos [En línea] http://ingenieriadelaseguridad.blogspot.com.co/p/control-de-accesos.html [Citado el 22 de mayo de 2017]

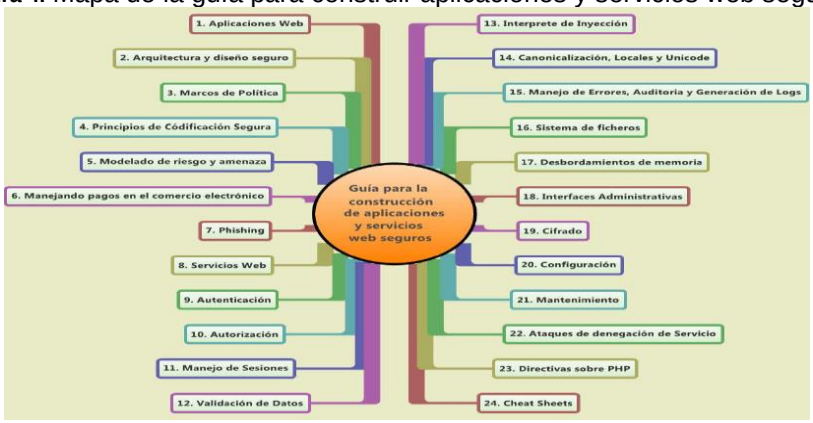
2.1.4 Seguridad en aplicaciones Web

"La organización *Open Web Application Security Project (OWASP*) elaboró una guía para construir Aplicaciones Web Seguras y Servicios Web Seguros. En resumen, desde las vulnerabilidades más antiguas como la Inyección SQL, hasta las más actuales como suplantación de identidad, sesiones, el cumplimiento de reglas y cuestiones de privacidad. Esto con el objetivo de ayudar a los desarrolladores, revisores de código, arquitectos de Software, entre otros, a tener pautas para evitar éstos problemas en el desarrollo, como otros mecanismos para hacer de las aplicaciones web más seguras"[10].

Principalmente la guía tiene en cuenta seguridad en aplicaciones Web y servicios, con ejemplos en los lenguajes de programación: J2EE, ASP.NET, PHP.

La guía para construir Aplicaciones y servicios Web Seguros está compuesta por los siguientes ítems:

Figura 4. Mapa de la guía para construir aplicaciones y servicios web seguros



Fuente: https://www.owasp.org/images/b/b2/OWASP_Development_Guide_2.0.1_Spanish.pdf

2.1.5 Auditoria de seguridad

Previo a definir que es una auditoría de seguridad primero se va a tratar de definir qué se entiende por auditoría y por seguridad:

10 OWASP Fundación, guía para construir aplicaciones y servicios web seguros [En línea], https://www.owasp.org/images/b/b2/OWASP_Development_Guide_2.0.1_Spanish.pdf, [Citado el 12 de septiembre de 2011]

“Una auditoría es la evaluación o revisión de las cualidades de un sistema, persona, objeto, proceso, producto, etc., para saber cómo se posiciona éste de acuerdo a un marco de referencia o una serie de criterios”[11].

Según la Real Academia Española (RAE) seguridad es la cualidad de seguro. Y seguro es todo aquello libre y exento de todo peligro, daño o riesgo[12].

Así pues, se podría entender por auditoría de seguridad aquel proceso que una vez llevado a cabo permite evaluar e identificar de forma sistemática el estado de la seguridad (en este caso de un aplicativo web) en relación a una serie de criterios o normas.

Mediante este proceso se tratará de identificar aquellos riesgos que pudieran afectar a la confidencialidad, integridad y/o disponibilidad de un aplicativo y de los sistemas asociados con éste, identificando vulnerabilidades.

2.2 MARCO LEGAL

2.2.1 Ley 1273 de 2009

“Los tres principios fundamentales de la seguridad son la confidencialidad, la integridad y la disponibilidad como se citó anteriormente. Para preservar estos principios el Congreso de Colombia aprobó la Ley 1273 de 2009, que pretende proteger la información, los datos y la preservación integral de los sistemas que utilicen tecnologías de la información y las comunicaciones”[13].

La ley en su primer capítulo tiene en cuenta los siguientes artículos, que son los más esenciales y directos con la información:

- ✓ Acceso abusivo a un sistema informático.
- ✓ Obstaculización ilegítima de sistema informático o red de telecomunicación.
- ✓ Interceptación de datos informáticos.

---

[11] PARRA, ANDRES [En línea] https://www.gerencie.com/auditoria-de-sistemas-de-informacion.html, [Citado el 26 de mayo de 2017]

[12] Academia española [En línea] http://dle.rae.es/srv/fetch?id=XTrIaQd [Citado el 25 de mayo de 2017]

[13] Secretaria del senado, Ley 1273 de 2009 [En línea], http://www.secretariasenado.gov.co/basedoc/ley/2009/ley_1273_2009.html, [Citado el 20 de febrero de 2016]

- ✓ Daño informático.
- ✓ Uso de software malicioso.
- ✓ Violación a datos personales.
- ✓ Suplantación de sitios web para capturar datos personales.
- ✓ Circunstancias de agravación punitiva.

Esta ley es de gran importancia como un apoyo legal para proteger la información de las organizaciones o personas, ya que no están exentas a estos problemas.

Además, conocer las multas y penas de estas violaciones, posibilitan un mecanismo de respuesta rápido de las organizaciones o personas para defender su activo más importante que es la información.

3 METODOLOGIA

Este proyecto se desarrolló en tres fases, la primera consistió en una investigación bibliográfica, la segunda en un estudio de caso para conocer C&M CONSULTORES y cómo está organizada desde el punto de vista informático, y la tercera en el diseño de un plan detallado de auditoría a las aplicaciones Web de la empresa.

Para comprender las vulnerabilidades, amenazas y riesgos de las aplicaciones Web, de igual manera que para poder analizar los procedimientos, pruebas y herramientas de software libre disponibles para auditoría, se procedió a hacer una búsqueda de fuentes de información sobre estos temas; después de identificar varias fuentes entre libros, artículos de revista y tutoriales se seleccionaron aquellos que ofrecían información más confiable y completa para hacer un estudio más profundo y detallado que permitiera extraer información suficiente para alcanzar los objetivos del proyecto.

La segunda fase se orientó al conocimiento de la empresa C&M CONSULTORES, a identificar las características particulares que deben ser tenidas en cuenta en una auditoría, ya que las metodologías disponibles consideran las organizaciones en general, con aquellos aspectos que les son comunes, pero todas las organizaciones no funcionan de igual forma y para que la auditoría ofrezca resultados confiables y útiles es necesario que se adapte a las circunstancias de la organización a que se aplican.

Con base en el conocimiento de las dos fases anteriores se procedió a revisar las metodologías de auditoría más utilizadas y extraer de ellas los elementos que se consideraron adecuados para una auditoría a la empresa C&M CONSULTORES y de igual manera se revisaron las herramientas para identificar las más convenientes en cada paso.

Por último, se realizó un estudio técnico sobre la empresa y se presentó una propuesta de auditoria de seguridad para aplicaciones Web.

El área de Informática de C&M Consultores se encarga de la recolección, manejo y almacenamiento de datos de la misma, ya sea hacia el mundo externo de la empresa o la parte interna de la misma.

C&M Consultores utiliza como su principal Aplicación Web, *SharePoint* con sus funciones de *Office* y servicios avanzados para mensajería, uso compartido de documentos, cumplimiento y características de administración para TI. Por lo tanto, resulta fácil auditar esta aplicación Web, ya que se puede usar los

siguientes informes de registro de auditoría proporcionados para ayudar a determinar quién realiza una acción determinada con el contenido de una colección de sitios:

- Revisión del contenido: Informa sobre los usuarios que han visto el contenido en un sitio.
- Modificaciones del contenido: Notifica los cambios que se realizan en el contenido, como modificar, eliminar o proteger y desproteger documentos.
- Eliminación: Notifica qué contenido ha sido eliminado.
- Tipo de contenido y modificaciones de lista: Notifica las adiciones, ediciones y eliminaciones en los tipos de contenido.
- Modificaciones de directiva: Notifica sobre los eventos que modifican las directivas de administración de información de la colección de sitios.
- Expiración y disposición: Notifica sobre todos los eventos relacionados con el modo en que se quita el contenido una vez que expira.
- Configuración de auditoría: Notifica sobre los cambios realizados en la configuración de auditoría.
- Configuración de seguridad: Notifica sobre los cambios realizados en la configuración de seguridad, como los eventos de usuario o grupo y los eventos de roles y derechos.
- Generar un informe personalizado: Puede especificar los filtros para un informe personalizado; por ejemplo, puede limitar el informe a un conjunto específico de eventos, a elementos en una lista en particular, a un intervalo de fechas determinado o a eventos realizados por usuarios concretos.

El resultado de este, servirá para redactar y sacar conclusiones, que conlleven a la empresa a tomar las mejoras continuas en sus desarrollos tecnológicos en los momentos que se quiera a dar a nueva implementaciones o aplicaciones a la vanguardia de la tecnología (servicios).

4 RESULTADOS

4.1 VULNERABILIDADES, AMENAZAS Y RIESGOS DE LAS APLICACIONES WEB

4.1.1 Vulnerabilidades de las aplicaciones Web

Son todos aquellos problemas de seguridad que afectan las páginas web, por lo general estos problemas permiten modificación y extracción de la información, lo cual es muy grave para las organizaciones. La mayoría de éstos son registrados por medio de un identificador de CVE (*Common Vulnerabilty Exposure*), él cual es un diccionario de los problemas de seguridad encontrados en Internet.

A continuación, se hablará de algunas de ellas:

- ***CROSS SITE SCRIPTING (XSS):***

"Es una técnica Hacking que permite a un atacante explotar vulnerabilidades en aplicaciones web e inyectar scripts del lado del cliente en éstas. Un ataque exitoso puede permitir al atacante secuestrar sesiones de usuario, robar información sensible o cambiar información en el sitio web"[14].

Hay dos tipos principales de XSS:

- ✓ No persistente o reflejado
- ✓ Persistente o almacenado

El **no *persistente* o *reflejado***. Este ataque es uno de los más comunes, la raíz de la vulnerabilidad es el manejo inapropiado (falta de validación) de solicitudes de datos HTTP por el código del servidor, permitiendo a los sitios maliciosos reflejar código malicioso y atacar a otros usuarios. El principal vector de ataque es usualmente un mensaje de correo que contiene una URL maliciosa, cuando el

[14] PEREZ, Ignacio Vulnerabilidad XSS [En línea] https://www.welivesecurity.com/la-es/2015/04/29/vulnerabilidad-xss-cross-site-scripting-sitios-web/ [Citado el 02 de febrero de 2017]

usuario da clic en la URL, el código malicioso es ejecutado. Esta vulnerabilidad aprovecha el concepto de arquitectura cliente servidor (Servidor WEB Navegador Web), el navegador ejecuta el código porque cree que es el código original y no uno alterado.

El ***persistente* o *almacenado***. Este ataque no requiere usuarios que den clic en una URL con el fin de ejecutar código malicioso. En este caso el código es capaz de vivir en el servidor vulnerable y está embebido en el código HTML. Una vez más, este tipo de ataque es el resultado directo de validaciones pobres en el lado del servidor, lo que permite forzar entradas maliciosas que pueden ser mostradas en el sitio web. Este tipo de ataque es particularmente riesgoso, no solo porque no requiere una intervención directa del usuario sino porque tiene un alcance global más peligroso.

- ***INYECCIÓN DE CÓDIGO SQL (SQL Injection):***

Esta vulnerabilidad surge de las malas prácticas de programación en las solicitudes HTTP (POST y GETs), así un atacante aprovecha el mal manejo de éstas, inyectando código SQL adicional, por ejemplo:

sSql = "SELECT LocationName FROM Locations" + "WHERE Location|D =" + Request ["Location|D"];

La variable Request ["Location|D"]; puede recibir consultas SQL y el atacante puede aprovecharse para modificar la consulta y sacar información.

- ***EJECUCIÓN DE COMANDOS (Command Execution):***

Este tipo de vulnerabilidad toma ventaja de la falta de validación en las entradas en un sitio web, donde el atacante puede correr comandos del sistema operativo en la aplicación web vulnerada. Generalmente, esta vulnerabilidad permite aprovechar, que los datos de usuario son pasados como parámetros a operaciones de entrada y salida, para así añadir comandos de sistema operativo por medio de caracteres especiales.

- ***DESBORDAMIENTO DE BUFFER (Buffer Overflow):***

"Es un ataque que ocurre cuando un usuario malicioso sobrecarga la memoria del sistema temporal (llamada buffer) para causar estragos en la máquina de la víctima. A menudo, los atacantes también incluyen código de instrucción para aprovechar más la vulnerabilidad, como por ejemplo ejecutar código malicioso, acceder o modificar datos confidenciales o incluso enviar información al atacante"[15].

- ***DENEGACIÓN DE SERVICIO (DoS):***

Es un tipo de vulnerabilidad que permite a un atacante agotar los recursos informáticos de un sistema, por medio de millones de solicitudes, agotando recursos como CPU, Memoria, acceso a la red, que imposibilitan el acceso a dicho sistema.

"Este ataque tiene una variante llamada Ataque de Denegación de Servicio Distribuido (DDoS), en el cual varios computadores infectados con virus atacan el servidor objetivo desde muchos lugares"[16].

- ***CROSS-SITE REQUEST FORGERY (CSRF):***

Es un ataque que presiona a la víctima a cargar una página que contiene una solicitud maliciosa, es maliciosa en el sentido en que hereda la identidad y privilegios de la víctima para ejecutar acciones no deseadas en ella, por ejemplo, cambiar la dirección de correo de la víctima, la dirección del hogar, o la contraseña.

---

[15] PEREZ, Ignacio Vulnerabilidad XSS [En línea] https://www.welivesecurity.com/la-es/2015/04/29/vulnerabilidad-xss-cross-site-scripting-sitios-web/ [Citado el 02 de marzo de 2017]

[16] ROUSE, Margaret [En línea] http://searchdatacenter.techtarget.com/es/definicion/Ataque-de-denegacion-de-servicio [Citado el 02 de marzo de 2017]

- ***INCLUSIÓN REMOTA DE ARCHIVOS (Remote File Inclusion):***

Es una técnica de ataque usada para explotar “la inclusión dinámica de librerías”, por ejemplo:

```
$incfile = $_REQUEST["pag"];

include($incfile.".php");
```

Como se puede ver desde la URL: http://ejemplo.com/?pag=pagina1.php

El valor asignado a pag es pagina1.php, pero imaginemos si en lugar de pagina1.php hubiera un link hacia otro sitio que tenga una web Shell, algo como:

http://ejemplo.com/?pag=www.shellremota.com

Lo cual posibilitaría la ejecución de comandos, modificación del sitio, entre otros.

- ✓ **Diferentes enfoques para realizar un análisis de vulnerabilidades**

Una de las diferentes partes de la que consta una auditoría es el análisis o evaluación de vulnerabilidades que se podrá enfocar utilizando diversas estrategias. Los resultados variarán considerablemente dependiendo de estos enfoques:

- **Black box (Caja negra):** Para este tipo de análisis los auditores se ponen en la piel de un atacante el cual no tienen conocimiento alguno de la aplicación o de los sistemas asociados a ésta.
- **White box (Caja blanca):** En los análisis de caja blanca se presenta a los auditores con todo el conocimiento del aplicativo, así como con una copia del código fuente para que la revisen a conciencia.
- **Grey box (Caja gris):** Este enfoque consiste en una mezcla de los dos anteriores. En esta ocasión se facilitará a los técnicos una parte de información acerca del funcionamiento de la aplicación y de los sistemas

con los que interactúa y con ella procederán a realizar el análisis de vulnerabilidades.

Si se invierte el mismo tiempo en hacer la misma auditoría con cada uno de los tres enfoques se observarán una serie de diferencias.

El enfoque más realista para realizar la evaluación de vulnerabilidades será el de caja negra, puesto que los auditores tendrán el mismo conocimiento que el que pudiera tener un posible atacante. Se deberá por lo tanto entender que éste podría generar algún que otro falso positivo o no encontrar ninguna vulnerabilidad aun existiendo éstas.

La revisión más exhaustiva y por lo consiguiente fiable sería la de caja blanca, pero, sin embargo, ésta también será la más laboriosa y costosa económicamente. Las auditorías de caja blanca suelen incluir una revisión de código con la que los auditores tendrán total conocimiento sobre la aplicación y podrán identificar de forma sencilla los puntos más críticos de esta o si por ejemplo se repiten ciertos patrones de fallos que pudieran derivar en vulnerabilidades. Con este enfoque y debido a la complejidad que entraña es posible que no se pueda finalizar a tiempo la evaluación si la ventana acordada para la evaluación no es lo suficientemente grande para el alcance definido.

Por último, si se enfoca el análisis de vulnerabilidades con una metodología de caja gris, se conseguiría que los auditores estuvieran en una posición aproximada a la de un atacante, pero con la ventaja de que pueden identificar con más facilidad aquellas partes más críticas o que requieran más atención.

Se debe por lo tanto entender que el hecho de elegir entre un enfoque u otro variará dependiendo de los intereses y presupuesto del que se disponga.

El haber concluido una auditoria, no es garantía de que la aplicación sea 100% segura.

Se debe comprender que no existe un producto totalmente seguro y que siempre hay riesgos. De igual modo se debe observar que en una auditoría de seguridad es posible que no se identifiquen todos los fallos que hay en la aplicación web. Hay una serie de factores que podrían afectar la precisión de los resultados:

- ✓ Presupuesto: la realización de una auditoría lo suficientemente rigurosa se podría ver afectada por el impacto económico que tendría ésta sobre el coste total del proyecto.

- ✓ Tiempo: éste es un factor que generalmente, aunque no necesariamente, está relacionado con el punto anterior. No será posible identificar todas las vulnerabilidades existentes en el sistema o aplicativo debido a una limitación de tiempo, que puede venir impuesta por unas limitaciones en el presupuesto

- ✓ Alcance: por diversos motivos puede haber parte de la aplicación o sistemas asociados con ésta que queden fuera del alcance de la auditoría dejando, por lo tanto, parte de ella sin auditar.

- ✓ Factores externos: aun siendo un software de desarrollo propio no es raro en encontrarse con el uso de librerías de terceros sobre las que no se tiene control. En caso de ser estas vulnerables, nuestra aplicación también podría verse afectada.

Es evidente entonces realizar un análisis de vulnerabilidades no garantiza que la aplicación evaluada sea segura. Debe quedar claro que no hay aplicación 100% segura y que una auditoría sólo sirve para garantizar unos mínimos o que se ha expresado un interés en proteger el aplicativo.

- ✓ **Resultados**

Así pues, la auditoría de seguridad formará parte del último paso en la implementación de medidas defensivas, servirá para evaluar la efectividad de éstas y como resultado generará un informe en el que se detallarán las vulnerabilidades encontradas y posiblemente una serie de recomendaciones para solventarlas.

Si el informe de resultados no es favorable habrá dos opciones para cada una de las vulnerabilidades encontradas: la primera sería aceptar el riesgo y la segunda solventar el fallo. Una vez finalizado este proceso se deberá volver a evaluar otra vez si se han corregido apropiadamente. Esta parte debe moverse al marco teórico, al tema de auditoría.

- ✓ **Como identificar una vulnerabilidad**

- Verificando que se separe la información no confiable del comando o consulta, usando variables parametrizadas en todas las sentencias preparadas y procedimientos almacenados, evitando las consultas dinámicas.
- Utilizar herramientas de análisis de códigos.

- Análisis dinámico automatizado en donde se detecta el ataque mediante manejo de errores

4.1.2 Amenazas de las aplicaciones Web

"Se entiende como AMENAZA todas aquellas máquinas, personas y/o sucesos que atacan a un sistema causando daño"[17].

Hay diferentes tipos de Amenazas como son:

- Amenazas físicas
- Amenazas Ambientales
- Amenazas de Software Malicioso
- Amenazas de Robo
- Amenazas de Destrucción o modificación de la Información
- Amenaza de Errores los cuales pueden ser intencionados o No intencionados

Las amenazas se clasifican de 2 maneras, según su origen y según el daño

- **Según el origen:**

***Accidentales*:** En este grupo se incluyen: los incendios, los fallos en los equipos, en redes, sistemas operativos o en software, las inundaciones y los errores humanos.

***Intencionadas*:** Estas amenazas son las consecuencias de las acciones humanas y su origen puede radicarse a nivel intra o extra organizacional. Como ejemplo tenemos: la inyección de software malicioso, intrusión informática, robo o hurto.

Este tipo de amenaza puede ser originada desde fuera de la empresa o dentro de la misma organización.

- **Según el daño:**

Interrupción: Hace énfasis en cuanto a la deshabilitación del acceso a la información. Su eje central es la destrucción de los componentes físicos como el disco duro; saturando los canales de comunicación o bloqueando el acceso a los datos.

---

17 DEVELOPER, NETWORK [Disponible en] https://msdn.microsoft.com/es-es/library/f13d73y6(v=vs.100).aspx, [Citado el 18 de mayo de 2017]

Interceptación: Es el acceso denegado a un determinado recurso del sistema con tal fin que se pueda captar información confidencial de la organización.

Modificación: Tiene 2 acciones: acceder a una información y modificar dicha información.

Fabricación: Añade información falsa en la información del sistema.

Hay que tener en cuenta que para poder realizar un análisis de riesgo en un sistema de información es necesario lo siguiente:

- ✓ Ejecutar un proceso secuencial de análisis de activos.
- ✓ Identificar las vulnerabilidades.
- ✓ Identificar y valorar las amenazas.
- ✓ Identificar las medidas de seguridad existentes.
- ✓ Identificar los objetivos de seguridad de la información en la organización.
- ✓ Determinar la medición de los riesgos, el impacto del ataque.
- ✓ Seleccionar las medidas de protección.

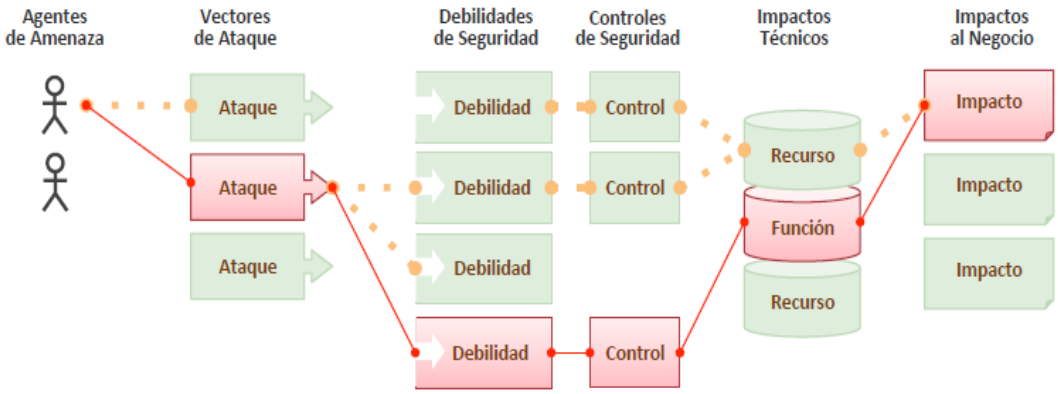
4.1.3 Riesgos de las aplicaciones Web

"Los atacantes pueden potencialmente usar rutas diferentes a través de la aplicación para hacer daño a su negocio u organización. Cada una de estas rutas representa un riesgo que puede o no ser lo suficientemente grave como para justificar la atención"[18].

---

18 OWASP, HERRAMIENTAS, Herramientas OWASP [Disponible en] https://www.owasp.org/images/2/2d/OWASP_Top_10_-_2010_FINAL_%28spanish%29.pdf

Figura 5. Arquitectura cliente servidor



Fuente: https://www.owasp.org/images/5/5f/OWASP_Top_10_-_2013_Final_-_Espa%C3%B1ol.pdf.

A veces, estas rutas son difíciles de encontrar y explotar. Del mismo modo, el daño que se causa puede ir de ninguna consecuencia, o ponerlo fuera del negocio.

- **Gestión de riesgo en la seguridad informática**

La Gestión de Riesgo es un método para determinar, analizar, valorar y clasificar el riesgo para posteriormente implementar mecanismos que permitan controlarlo.

La Gestión de Riesgo se divide en cuatro Partes:

- ✓ Análisis
- ✓ Clasificación
- ✓ Reducción
- ✓ Control

Las cuales se citan a continuación:

Figura 6. Gestión de Riesgos

Fuente: http://www.msal.gob.ar/salud-y-desastres/index.php/informacion-para-comunicadores

Análisis: En la parte de análisis se determina los componentes del sistema que requieren protección, las vulnerabilidades que lo debilitan y las amenazas que lo ponen en peligro, con el objetivo de revelar su grado de riesgo. Por lo tanto, al identificar las vulnerabilidades y las amenazas del sistema, permitirá conocer los riesgos potenciales que atentan contra la seguridad del sistema.

Clasificación: En la parte de Clasificación se determina si los riesgos encontrados y los riesgos restantes son aceptables.

Reducción: En la parte de Reducción se define e implementa las medidas de protección y de igual forma se sensibiliza y capacita a los usuarios conforme a las necesidades.

Control: En la parte de Control se analiza el funcionamiento, la efectividad y el cumplimiento de las medidas y si es el caso ajustarlas.

- **Análisis de riesgos**

Cuando se quiere dotar de seguridad a un sistema informático es necesario determinar los elementos o activos que requieren protección, identificar el nivel de vulnerabilidad de cada uno frente a determinadas amenazas y valorar el impacto que un ataque ocasionaría sobre el sistema informático.

“Los activos son los recursos que pertenecen al sistema de información, se pueden clasificar en:

***Datos*:** Los datos son el núcleo (Core) de la organización, toda empresa u organización depende de sus datos y éstos pueden ser: Económicos, fiscales, recurso humano, clientes o proveedores.

***Software*:** Son el conjunto de aplicaciones instaladas que se encuentran en los equipos, que hacen parte del sistema de información, estas aplicaciones reciben, gestionan o transforman los datos.

***Hardware*:** Conjunto de equipos (Servidores y Terminales) que contienen las aplicaciones y permiten su funcionamiento, de igual manera almacenan los datos del sistema de información.

***Redes*:** Representa la vía de comunicación y transmisión de datos. La red informática nombra al conjunto de computadoras y otros equipos interconectados, que comparten información, recursos y servicios. Puede a su vez dividirse en diversas categorías, según su alcance (red de área local o LAN, red de área metropolitana o MAN, red de área amplia o WAN, etc.), su método

de conexión (por cable coaxial, fibra óptica, radio, microondas, infrarrojos) o su relación funcional (cliente-servidor, persona a persona), entre otras"[19].

***Soportes*:** Son los lugares donde la información queda registrada y almacenada durante un periodo de tiempo o de manera permanente: Tarjetas de memoria, Discos duros, DVD, CD.

***Personal*:** Está conformado por las personas que interactúan con el sistema de información: Programadores, Administradores, Usuarios internos y externos. Estudios calculan que se producen más fallos de seguridad por intervención humana que por los fallos de software.

- **Riesgos más importantes en las aplicaciones Web**

A1 – Inyección: Los diferentes inconvenientes de inyección como SQL, OS, y LDAP, ocurren cuando datos no confiables son enviados a un intérprete como parte de un comando o consulta. Los datos hostiles del atacante pueden engañar al intérprete en ejecutar comandos no intencionados o acceder datos no autorizados.

"El problema de este tipo de ataque, es que cualquier persona lo puede llevar a cabo ya que consiste únicamente en el envío de cadenas de texto a la aplicación, generalmente a través de métodos genuinos proporcionados por la aplicación para recibir datos válidos. La efectividad del ataque ya dependerá de la habilidad y conocimientos del atacante. El impacto de este ataque puede ir desde la simple consulta de datos almacenados, hasta el control de nuestro sistema o servidores.

Generalmente, no se habla solo de inyección SQL, debido a que también hay inyección LDAP, inyección XPATH, comandos del sistema operativo, argumentos de programas, etc…[20]"

Inyección SQL

La inyección de código SQL es una técnica de ataque usada para explotar sitios web que construyen sentencias SQL a partir de entradas facilitadas por el

---

[19] Que son las redes informáticas [En línea], http://www.redusers.com/noticias/que-es-una-red-informatica/, [Citado el 5 de mayo de 2016]

[20] OWASP [En línea], https://infow.wordpress.com/2010/12/17/owasp-top-ten-a1-inyeccion/ [Citado el 5 de mayo de 2016]

usuario. El resto de inyecciones son similares, en los argumentos de un programa o en un sistema operativo, si no se filtran las cadenas que se introducen en él, se pueden realizar acciones para las que la aplicación no está preparada.

Inyección LDAP

La inyección LDAP es una técnica de ataque usada para explotar sitios web que construyen sentencias LDAP a partir de datos de entrada suministrados por el usuario.

Inyección Xpath

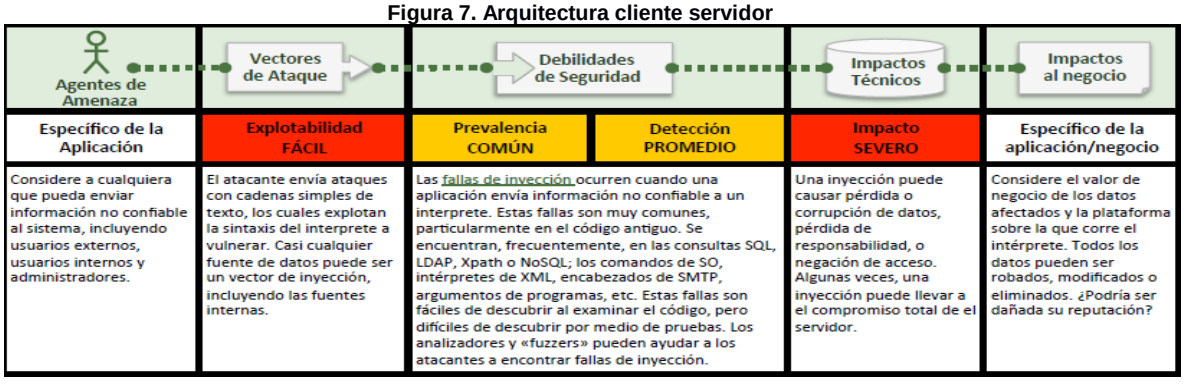
La inyección XPath es una técnica de ataque utilizada para explotar sitios web que construyen consultas Xpath con datos de entrada facilitados por el usuario.

Inyección de código SSI

La inyección de código SSI (Server-side Include) es una técnica de explotación en la parte servidora que permite a un atacante enviar código a una aplicación web, que posteriormente será ejecutado localmente por el servidor web.

Para evitar las inyecciones del tipo que sean, lo que debo hacer es filtrar cualquier tipo de dato que se introduzca en nuestra aplicación. Existen herramientas de análisis que nos pueden ayudar a auditar nuestro código, pero algo que deberíamos hacer desde ya, es mentalizar a nuestros desarrolladores sobre la importancia de la validación y filtrado de datos.

Figura 7. Arquitectura cliente servidor



Fuente: https://es.wikipedia.org/wiki/Inyecci%C3%B3n_SQL

A2 – Secuencia de comandos en sitios cruzados (XSS): “Las fallas XSS ocurren cada vez que una aplicación toma datos no confiables y los envía a la web sin una validación y codificación apropiada. XSS lo que permite a los atacantes es ejecutar secuencia de comandos en el navegador de la víctima los cuales pueden secuestrar las sesiones de usuario, destruir sitios web, o dirigir al usuario hacia un sitio malicioso.

Esta vulnerabilidad se produce cuando el sistema envía datos suministrados por los usuarios, al navegador, que no han sido validados, de forma un usuario malicioso pueda ejecutar código malicioso en el servidor web, secuestrando la sesión”[21].

Un escenario posible sería cuando una aplicación utiliza datos no confiables en la construcción del siguiente código HTML sin validar o escapar los datos:

El atacante modifica el parámetro ‘CC’ en el navegador con el siguiente texto:

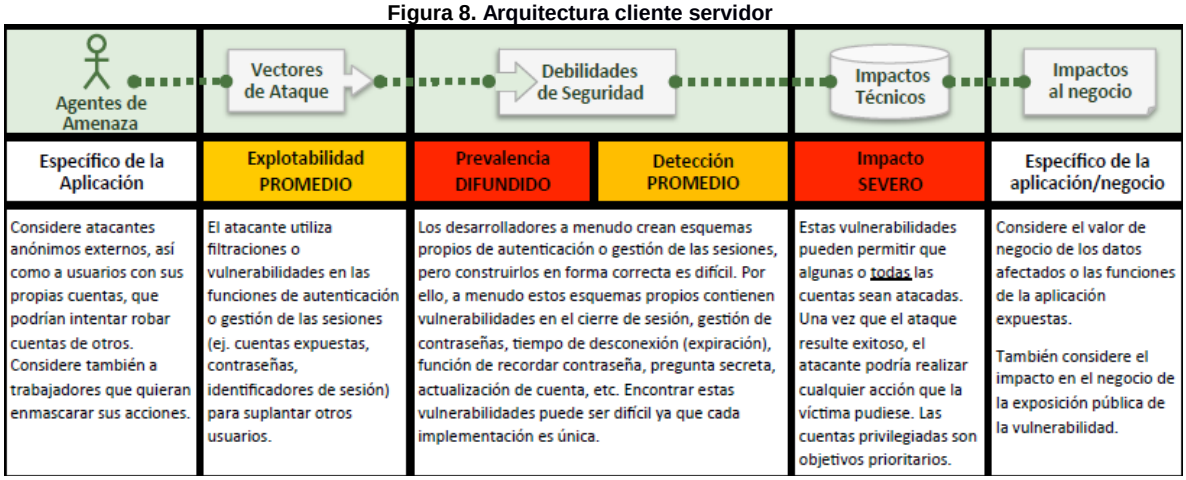
Lo que causa que el identificador de sesión de la víctima sea enviado al sitio web del atacante, permitiendo al atacante secuestrar la sesión actual del usuario.

Para eliminar o minimizar esta vulnerabilidad, pasaremos todos los datos por un filtro que limpie los caracteres especiales. No se ha identificado la necesidad de incluir los caracteres especiales en ningún escenario de uso de la aplicación, pero si así fuera necesario, utilizaríamos codificaciones y decodificaciones de los datos, seguidas de comprobaciones de longitud y formato, que identifiquen dicha entrada de datos cómo correcta.

---

21 VULNERABILIDAES, [Disponible es] http://www.securityartwork.es/2010/03/24/owasp-top-10-iii-perdida-de-autenticacion-y-gestion-de-sesiones/ [Citado el 02 de abril de 2016]

Figura 8. Arquitectura cliente servidor



Fuente: https://es.wikipedia.org/wiki/Inyecci%C3%B3n_SQL

A3 – Pérdida de Autenticación y Gestión de Sesiones: "Una referencia directa a objetos ocurre cuando un desarrollador expone una referencia a un objeto de implementación interno, tal como un fichero, directorio, o base de datos. Sin un chequeo de control de acceso u otra protección, los atacantes pueden manipular estas referencias para acceder datos no autorizados.

Esta vulnerabilidad es aplicable en el entorno de administrador, ya que la autenticidad del usuario administrador de garantiza mediante la gestión de la sesión. Para evitar el intento de robo autenticación, forzaremos que el sistema realice una autenticación mediante usuario, y *password* de los usuarios administradores, y proteja las credenciales de cuentas y los *tokens* de sesión, limitando el número de intentos de log-in con bloqueo de cuenta tras muchos intentos fallidos y caducidad de la sesión tras inactividad"[22].

22 TOKENS, Autenticación y gestión [Disponible en] http://www.securityartwork.es/2010/03/24/owasp-top-10-iii-perdida-de-autenticacion-y-gestion-de-sesiones/ [Citado el 05 de junio de 2016]

Figura 9. Arquitectura cliente servidor

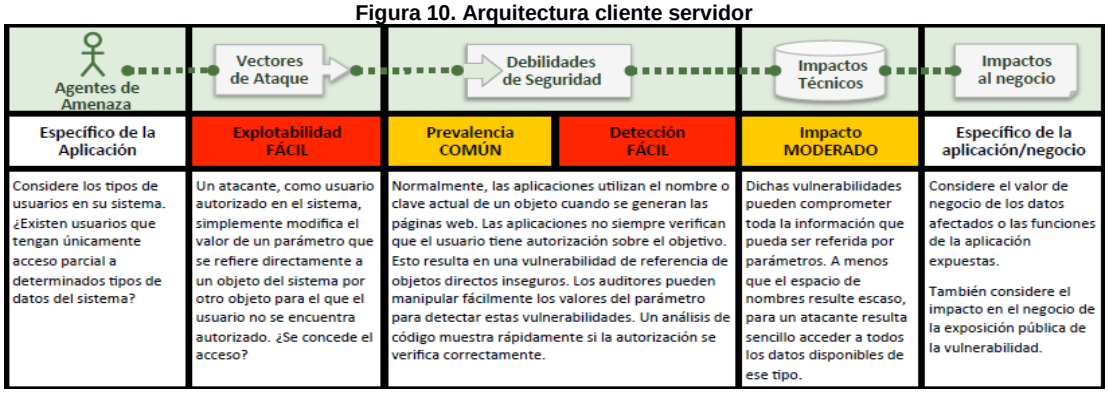
Agentes de Amenaza	Vectores de Ataque	Debilidades de Seguridad		Impactos Técnicos	Impactos al negocio
Específico de la Aplicación	Explotabilidad PROMEDIO	Prevalencia MUY DIFUNDIDA	Detección FACIL	Impacto MODERADO	Específico de la aplicación / negocio
Considere cualquier persona que pueda enviar datos no confiables al sistema, incluyendo usuarios externos, internos y administradores.	El atacante envía cadenas de texto que son secuencias de comandos de ataque que explotan el intérprete del navegador. Casi cualquier fuente de datos puede ser un vector de ataque, incluyendo fuentes internas tales como datos de la base de datos.	XSS es la falla de seguridad predominante en aplicaciones web. Ocurren cuando una aplicación, en una página enviada a un navegador incluye datos suministrados por un usuario sin ser validados o codificados apropiadamente. Existen tres tipos de fallas conocidas XSS: 1) Almacenadas, 2) Reflejadas, y 3) basadas en DOM. La mayoría de las fallas XSS son detectadas de forma relativamente fácil a través de pruebas o por medio del análisis del código.		El atacante puede ejecutar secuencias de comandos en el navegador de la víctima para secuestrar las sesiones de usuario, alterar la apariencia del sitio web, insertar código hostil, redirigir usuarios, secuestrar el navegador de la víctima utilizando malware, etc.	Considere el valor para el negocio del sistema afectado y de los datos que éste procesa. También considere el impacto en el negocio la exposición pública de la vulnerabilidad.

Fuente: https://es.wikipedia.org/wiki/Inyecci%C3%B3n_SQL

A4 – Referencia Directa Insegura a Objetos: "Un ataque CSRF obliga al navegador de una víctima autenticada a enviar una petición HTTP falsificado, incluyendo la sesión del usuario y cualquier otra información de autenticación incluida automáticamente, a una aplicación web vulnerable. Esto permite al atacante forzar al navegador de la víctima para generar pedidos que la aplicación vulnerable piensa son peticiones legítimas provenientes de la víctima"[23] .

Se evitará la exposición de referencias a objetos de implementación interna (por ejemplo, un archivo, un directorio, un registro de base de datos o una clave, como un URL o un parámetro de forma). Los atacantes podrían manipular estas referencias para acceder a otros objetos sin autorización.

Figura 10. Arquitectura cliente servidor



Agentes de Amenaza	Vectores de Ataque	Debilidades de Seguridad		Impactos Técnicos	Impactos al negocio
Específico de la Aplicación	Explotabilidad FÁCIL	Prevalencia COMÚN	Detección FÁCIL	Impacto MODERADO	Específico de la aplicación/negocio
Considere los tipos de usuarios en su sistema. ¿Existen usuarios que tengan únicamente acceso parcial a determinados tipos de datos del sistema?	Un atacante, como usuario autorizado en el sistema, simplemente modifica el valor de un parámetro que se refiere directamente a un objeto del sistema por otro objeto para el que el usuario no se encuentra autorizado. ¿Se concede el acceso?	Normalmente, las aplicaciones utilizan el nombre o clave actual de un objeto cuando se generan las páginas web. Las aplicaciones no siempre verifican que el usuario tiene autorización sobre el objetivo. Esto resulta en una vulnerabilidad de referencia de objetos directos inseguros. Los auditores pueden manipular fácilmente los valores del parámetro para detectar estas vulnerabilidades. Un análisis de código muestra rápidamente si la autorización se verifica correctamente.		Dichas vulnerabilidades pueden comprometer toda la información que pueda ser referida por parámetros. A menos que el espacio de nombres resulte escaso, para un atacante resulta sencillo acceder a todos los datos disponibles de ese tipo.	Considere el valor de negocio de los datos afectados o las funciones de la aplicación expuestas. También considere el impacto en el negocio de la exposición pública de la vulnerabilidad.

Fuente: https://es.wikipedia.org/wiki/Inyecci%C3%B3n_SQL

23 Referencia directa [Disponible en] http://www.securityartwork.es/2010/03/24/owasp-top-10-iii-perdida-de-autenticacion-y-gestion-de-sesiones/ [Citado el 02 de abril de 2016]

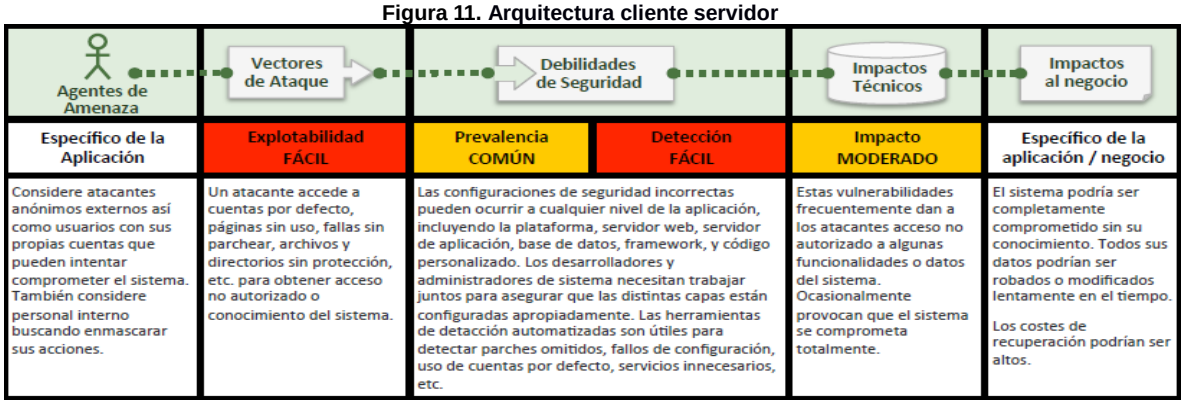
A5 – Falsificación de Peticiones en Sitios Cruzados (CSRF): “Una buena seguridad requiere tener definida e implementada una configuración segura para la aplicación, marcos de trabajo, servidor de aplicación, servidor *web*, base de datos, y plataforma. Todas estas configuraciones deben ser definidas, implementadas, y mantenidas ya que por lo general no son seguras por defecto.

Esto incluye mantener todo el software actualizado, incluidas las librerías de código utilizadas por la aplicación.

Un ataque de CSRF obliga al navegador de la víctima conectada a enviar una solicitud pre autenticada a una aplicación web vulnerable, que luego obliga al navegador de la víctima a ejecutar una acción para beneficio del atacante.

Con el objetivo de eliminar este riesgo, no se delegará la utilización de testigos en las credenciales de autorización ni en los *tokens* que presentan los exploradores web automáticamente, y se gestionarán las peticiones generadas por las aplicaciones mediante la inclusión de testigos no predecibles y únicos (al menos por cada sesión del usuario) en cada una de ellas”[24] .

Figura 11. Arquitectura cliente servidor



Fuente: https://es.wikipedia.org/wiki/Inyecci%C3%B3n_SQL

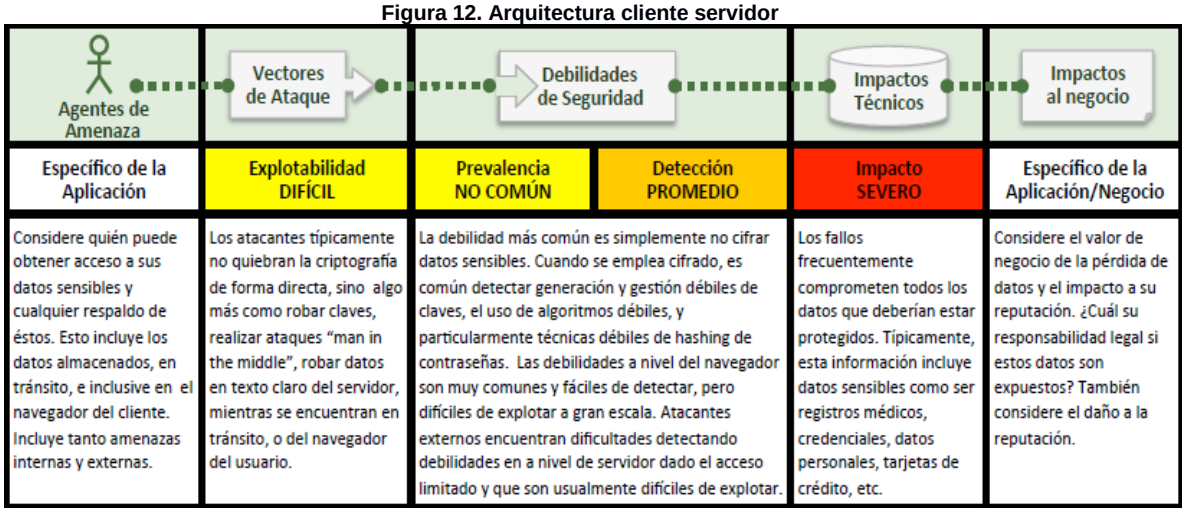
A6 – Defectuosa configuración de seguridad: Muchas aplicaciones web no protegen adecuadamente los datos sensibles, tales como tarjetas de crédito, *NSSs*, y credenciales de autenticación con mecanismos de cifrado o *hashing*. Atacantes pueden modificar o robar tales datos protegidos inadecuadamente para conducir robos de identidad, fraudes de tarjeta de crédito u otros crímenes.

---

24 Falsificación de peticiones [Disponible en] http://www.securityartwork.es/2010/03/24/owasp-top-10-iii-perdida-de-autenticacion-y-gestion-de-sesiones/ [Citado el 10 de abril de 2016]

“Esta vulnerabilidad consiste en aprovechar configuraciones por defecto o pobres de cualquiera de los elementos sobre los que funciona la aplicación web. La completa eliminación de este riesgo escapa del alcance del diseño del sistema ya que depende de otros factores”[25] .

Figura 12. Arquitectura cliente servidor



Fuente: https://es.wikipedia.org/wiki/Inyecci%C3%B3n_SQL

A7 – Almacenamiento Criptográfico Inseguro: “Muchas aplicaciones *webverifican* los privilegios de acceso a *URL* de generar enlaces o botones protegidos. Sin embargo, las aplicaciones necesitan realizar controles similares cada vez que estas páginas son accedidas, o los atacantes podrán falsificar *URL* para acceder a estas páginas igualmente.

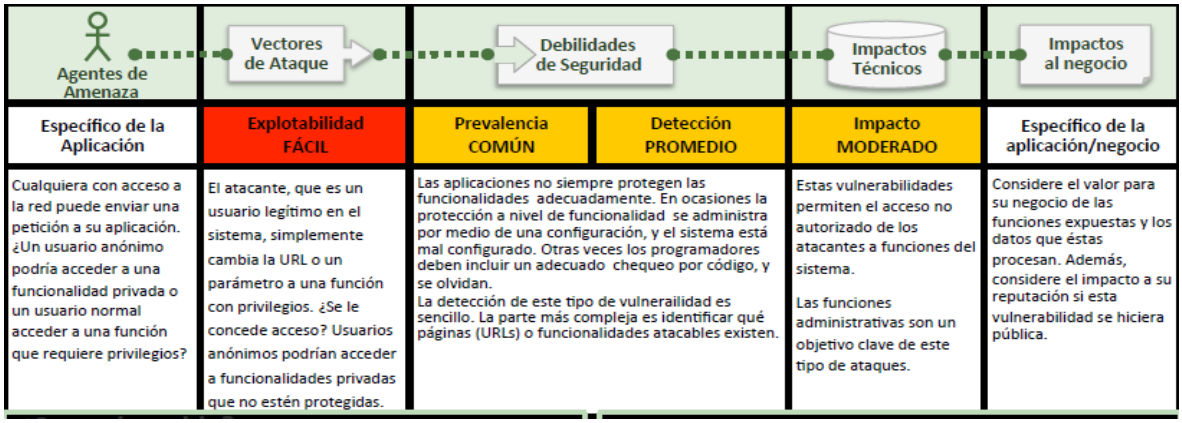
Proteger datos delicados con criptografía se ha convertido una parte clave de la mayoría de las aplicaciones Web. Simplemente no cifrar datos delicados está muy extendido. Aplicaciones que sí cifran, frecuentemente contienen criptografía mal diseñada, ya sea usando sistemas de cifrado no apropiados o cometiendo errores serios al usar algoritmos de cifrados sólidos. Estos defectos pueden conducir a la revelación de datos delicados y violaciones de cumplimiento de estándares”[26] .

---

25 Configuración de seguridad [Disponible en] http://www.sniferl4bs.com/2015/07/burpsuite-xiv-analisis-web-owasp_16.html

26 Almacenamiento Criptográfico [Disponible en] https://www.owasp.org/index.php/Top_10_2007-Almacenamiento_Criptogr%C3%A1fico_Inseguro

Figura 13. Arquitectura cliente servidor

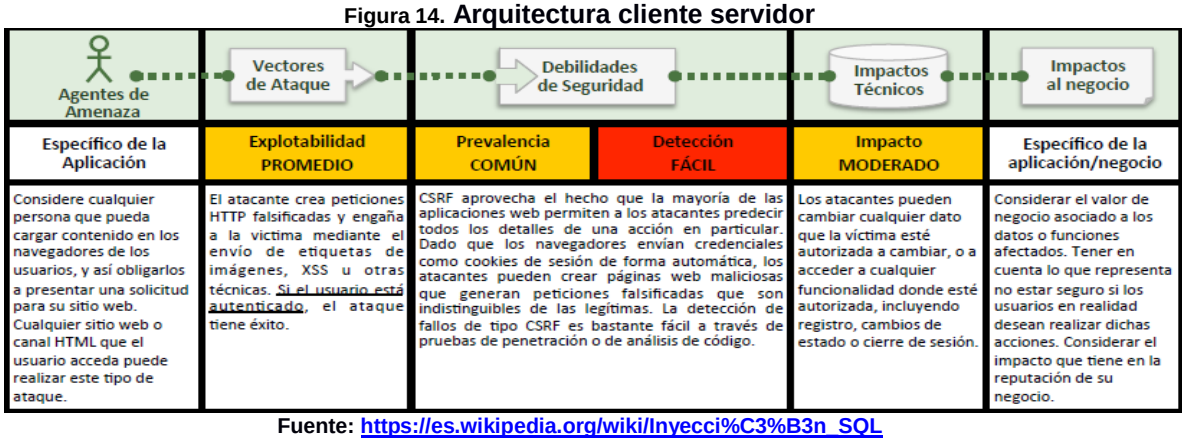


Fuente: https://es.wikipedia.org/wiki/Inyecci%C3%B3n_SQL

A8 - Falla de Restricción de Acceso a URL: "Las aplicaciones frecuentemente fallan al autenticar, cifrar, y proteger la confidencialidad e integridad de tráfico de red sensible. Cuando esto ocurre, es debido a la utilización de algoritmos débiles, certificados expirados, inválidos, o sencillamente no utilizados correctamente.

Para evitar que se realicen peticiones a la aplicación de zonas privadas o que requieran privilegios, y que se intente ganar acceso y realizar operaciones no autorizadas mediante el acceso a esas URL directamente, reforzaremos el control del acceso en la capa de presentación y la lógica comercial para todas las URL que lo necesiten. No se permitirá a los usuarios no autorizados ver vínculos o URL"[27] .

Figura 14. Arquitectura cliente servidor



Fuente: https://es.wikipedia.org/wiki/Inyecci%C3%B3n_SQL

27 Restricción de acceso a URL [Disponible en] https://www.owasp.org/index.php/Acknowledgements [Citado el 13 de abril de 2017]

A9 – Protección Insuficiente en la capa de Transporte: Las aplicaciones frecuentemente fallan al autenticar, cifrar, y proteger la confidencialidad e integridad de tráfico de red sensible. Cuando esto ocurre, es debido a la utilización de algoritmos débiles, certificados expirados, inválidos, o sencillamente no utilizados correctamente.

“Para hablar de la vulnerabilidad que nos concierne hoy, primero, me temo que voy a tener que hablar del modelo OSI. Supongo que todo aquel que tenga algún tipo de estudio relacionado con informática, redes, telecomunicaciones, etc…

El modelo OSI, es un modelo de interconexión de sistemas abiertos, o lo que es lo mismo en un lenguaje menos técnico, es la descripción de una propuesta de sistema de conexión entre dispositivos para que estos puedan interactuar entre sí a través de una red”[28] .

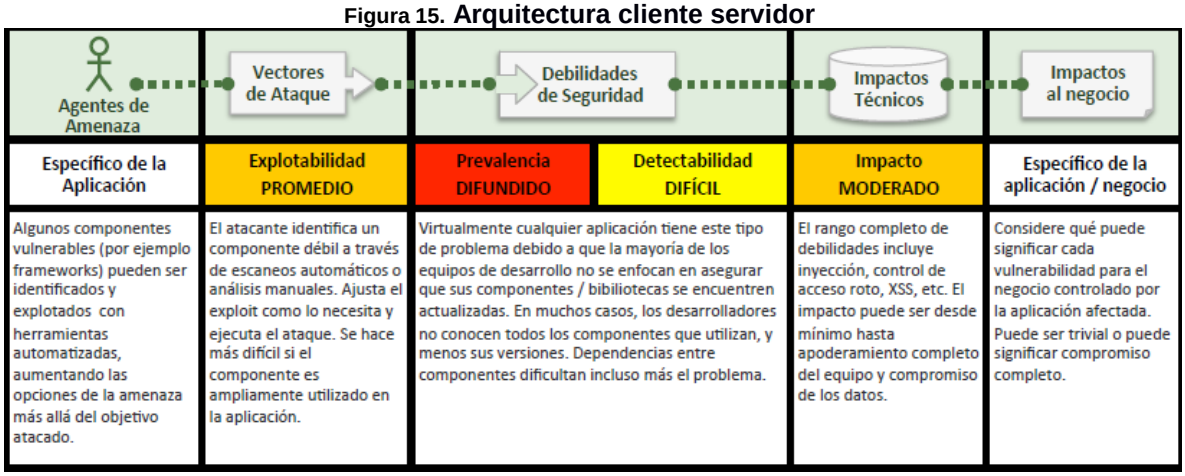
Por resumir un poco, el modelo OSI describe varias capas para realizar la comunicación, el cometido de cada una de estas diferentes capas y cómo podemos trabajar con ellas para finalmente obtener una comunicación correcta y entendible por parte de otros dispositivos.

Las capas que describe el modelo son siete:

- Capa física.
- Capa de enlace de datos.
- Capa de red.
- Capa de transporte.
- Capa de sesión.
- Capa de presentación.
- Capa de aplicación.

28 Protección Insuficiente en la capa de Transporte https://cyberacademy.deloitte.es/cs-faqs/10-vulnerabilidades-de-proteccion-insuficiente-en-la-capa-de-transporte-2/ [Citado el 23 de abril de 2017]

Figura 15. Arquitectura cliente servidor



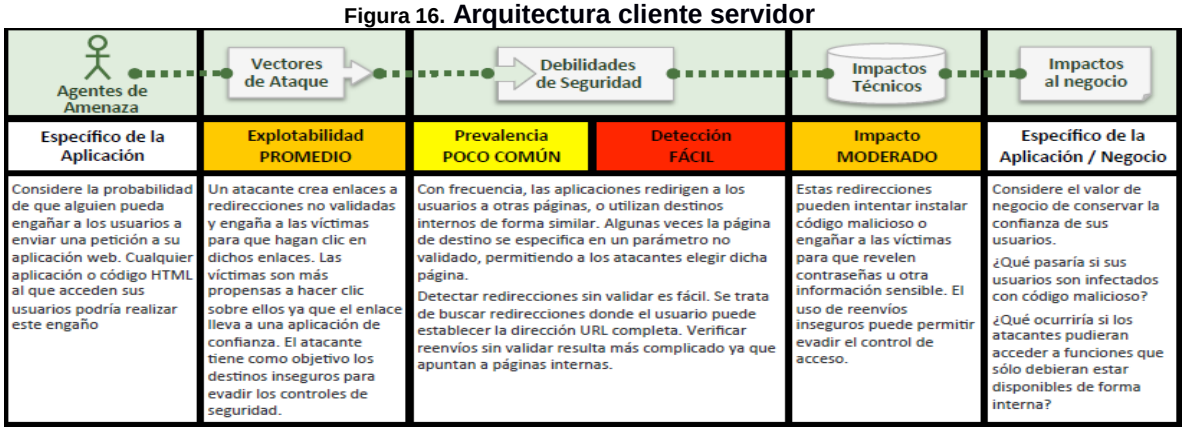
Fuente: https://es.wikipedia.org/wiki/Inyecci%C3%B3n_SQL

A10 - Redirecciones y Reenvíos no validados: "Las aplicaciones web frecuentemente me redirigen y reenvían hacia otras páginas o sitios web, y utilizan datos no confiables para determinar la página de destino. Sin una validación apropiada, los atacantes pueden redirigir a las víctimas hacia sitios de phishing o malware, o utilizar reenvíos para acceder páginas no autorizadas.

El uso por defecto de la aplicación no implica realizar re direccionamientos a sitios fuera del dominio de la aplicación en ningún escenario de uso. Para los re direccionamientos internos que se utilicen, no se involucrarán parámetros manipulables por el usuario para definir el destino, y en caso de ser necesario por mejorar la funcionalidad del sistema, se asegurará que el valor facilitado es validado y autorizado para el usuario, y se utilizará un valor de mapeo en lugar de la dirección"[29] .

29 Redirecciones y Reenvíos no validados [Disponible en] https://cyberacademy.deloitte.es/cs-faqs/10-vulnerabilidades-de-proteccion-insuficiente-en-la-capa-de-transporte-2/

Figura 16. Arquitectura cliente servidor



Fuente: https://es.wikipedia.org/wiki/Inyecci%C3%B3n_SQL

Luego de analizar los riesgos, es fundamental conocer los servicios que posee el sistema de información, y cuáles de estos servicios quedan a la intemperie para poder aplicarles los mecanismos de seguridad y así poder cumplir con los objetivos propuestos dentro de una organización

a. Plan de Contingencia

El Plan de Contingencia contiene las medidas preventivas y de recuperación frente a cualquier tipo de amenaza. Estas pueden ser de tres tipos:

- Plan de Respaldo: En el plan de respaldo se aplican medidas preventivas ante cualquier amenaza para evitar que se produzcan daños. Por ejemplo, Copias de respaldo.

- Plan de Emergencia: En el plan de emergencia se determina qué medidas tomar cuando se está materializando una amenaza o cuando acaba de producirse. Por ejemplo, restaurar las copias de seguridad.

- Plan de Recuperación: En el plan de recuperación se indican las medidas que se aplicarán cuando se ha producido un desastre. El objetivo principal es evaluar el impacto y regresar a un estado normal de funcionamiento del sistema.

b. Servicios de Seguridad

Entre los servicios de seguridad se encuentran:

- ***Integridad:*** Asegura los datos que posee el sistema de información no han sido alterados por personal no autorizado, así asegurando también que el contenido de los mensajes recibidos es el correcto.

- ***Confidencialidad:*** Brinda protección contra la revelación voluntaria o accidental de los datos en una comunicación.

- ***Disponibilidad:*** Permite que la información se encuentra accesible cuando la requiera personal autorizado.

- ***Autenticación:*** Verifica que un usuario que accede a un sistema de información es plenamente identificado.

- ***No repudio****:* No se podrá negar haber emitido o recibido una información cuando sí fue recibida.

- ***Control de acceso***: Solo los usuarios autorizados podrán acceder a los recursos del sistema.

c. **Mecanismos de seguridad**

Los mecanismos de seguridad se clasifican según la función que desempeñen, estos pueden ser Preventivos, Detectores o Correctores.

- ***Preventivos:*** Los mecanismos preventivos actúan antes de que se produzca un ataque, su misión principal es evitar el ataque.

- ***Detectores*****:** Los mecanismos detectores actúan cuando el ataque se ha efectuado y antes de que éste cause daños en el sistema.

- ***Correctores:*** Los mecanismos correctores actúan después de que se ha presentado un ataque y se haya producido daños. Su principal objetivo es el de corregir las consecuencias del daño.

Existen otros mecanismos de seguridad que dependen del sistema de información, de su función y de los riesgos a los que se expone el sistema, entre los cuales se encuentran los mecanismos de seguridad físicos y lógicos.
Los Mecanismos de Seguridad Físicos y Lógicos tienen como misión prevenir, detectar o corregir ataques al sistema, asegurando que los servicios de seguridad queden cubiertos.

- ❖ ***Seguridad Física***: Su objetivo es proteger al sistema de peligros físicos y lógicos. Un ejemplo de ellos son los dispositivos físicos de protección como los pararrayos, detectores de humo, cortafuegos por hardware, etc. Y por otro lado se encuentran las copias de respaldo o copias de seguridad de la información.

- ***Seguridad Lógica:*** Su principal objetivo es proteger digitalmente la información. A continuación, se citan algunos de ellos:

  - Control de acceso: Utilizando nombres de usuario y contraseña.

  - Cifrado de datos: Los datos se enmascaran utilizando algoritmos de encriptación. Fortalece la confidencialidad.

  - Antivirus: Estos detectan e impiden la entrada de virus y software malicioso. Protege la integridad de la información.

  - Cortafuegos: Los cortafuegos son dispositivos de software, hardware o mixtos que restringen el acceso al sistema. Protege la integridad de la información.

  - Firma digital: Es utilizada para la transmisión de mensajes telemáticos y en la gestión de documentos electrónicos. Protege la integridad y confidencialidad de la información.

  - Certificados digitales: Son documentos digitales que garantizan que una persona es quien dice ser. Protege la Integridad y confidencialidad de la información.

Las redes inalámbricas necesitan precauciones adicionales tales como el usar un SSID (*Service Set Identifier*) que no es otra cosa que darle un nombre a la red, preferiblemente un nombre que no llame mucho la atención. Otra precaución es proteger la red mediante claves encriptadas WPA (*Wifi Protected Access*). Y por último el Filtrado de direcciones MAC (*Media Access Control*), este es un mecanismo de acceso al sistema mediante hardware, el cual solo admite determinadas direcciones.

d. Política de Seguridad

La política de seguridad recopila los objetivos de la organización en materia de seguridad del sistema de información, los cuales se encuentran categorizados en cuatro grupos.

- El primero de ellos es identificar las necesidades de seguridad y los riesgos que amenazan el sistema de información y de igual forma evaluar el impacto frente a un eventual ataque.
- Tomar todas las medidas de seguridad que deban implementarse para afrontar los riesgos de cada activo.
- Determinar las reglas y los procedimientos que deben aplicarse para afrontar los riegos.

- ❖ d) Detectar todas las vulnerabilidades del sistema de información y controlar los fallos que se producen en los activos.
- ❖ Por último, definir un plan de contingencia.

e. Auditoria

La Auditoría es una prueba minuciosa de un sistema de información, que permite identificar y corregir vulnerabilidades en los activos que lo conforman y en los procesos que se conllevan.

La finalidad de la Auditoría es verificar que se cumplan los objetivos de la Política de Seguridad de la organización, así pues, proporciona una imagen real y actual del estado de seguridad de un sistema de información.

Luego de realizar una Auditoría, es decir, de realizar un análisis e identificar vulnerabilidades, el Auditor elabora un informe que debe contener una descripción de los activos y procesos analizados, una evaluación de las vulnerabilidades detectadas, una verificación del cumplimiento de la normatividad y una propuesta de medidas preventivas y correctivas.

"Existen herramientas para evaluar la seguridad en un sistema de información, éstas pueden ser manuales o software específico para auditoría. Con respecto a las Manuales, éstas pueden ser por observación directa de los activos, mediciones, cuestionarios, entrevistas, pruebas de funcionamientos entre otras.

La herramienta de software de Auditoría se le conoce por las siglas *CAAT* (*Computer Assisted Audit Techniques*), éstas ayudan a mejorar la eficiencia de una auditoría, ya que proporcionan una imagen total o parcial en tiempo real de un sistema de información emitiendo luego un informe de las vulnerabilidades encontradas"[30].

4.2 PROCEDIMIENTOS, PRUEBAS Y HERRAMIENTAS PARA AUDITAR APLICACIONES WEB

4.2.1 Procedimientos para auditar una aplicación Web

Los procedimientos para auditar una aplicación Web son:

- ➢ Revisión de la documentación de sistemas e identificación de los controles existentes.

---

30 Herramienta de Software de Auditoria http://www.microsoft.com/argentina/public/kit_base/licenciamiento/licsemgt/softaudt/tools.htm [Citado el 23 de abril de 2017]

- Entrevistas con los especialistas técnicos a fin de conocer las técnicas y controles aplicados.

- Utilización de software de manejo de base de datos para examinar el contenido de los archivos de datos.

- Técnicas de diagramas de flujo para documentar aplicaciones Web

Se debe tener en cuenta:

a. La documentación del sistema

Es importante revisar la situación en que se encuentra toda la documentación básica del sistema, como los manuales de procedimientos, de usuario y de análisis y si están acordes con las necesidades de la dependencia.

b. Operatividad del sistema

Se debe verificar que el sistema efectúe en cada etapa de su ciclo las especificaciones establecidas en el Análisis y Diseño y se cumpla con el procedimiento aprobado para el sistema. Debe analizarse y observar también, si realmente tiene los requisitos de operatividad que hagan al sistema eficiente y eficaz.

c. Ciclo de vida

Se debe revisar todo el ciclo del sistema:

- Generación del dato
- Ingreso del dato
- Transmisión del dato
- Procesamiento del dato

d. Actualización de archivos

Debe verificarse que existan registros y rutinas de control que con cierta frecuencia de tiempo o de período, determine si existe coherencia entre la cantidad de registros y valores de totales de los archivos.

e. Integridad de los datos

Debe, asimismo, en forma externa, establecerse procedimientos de comparación de la información producida por el sistema contra otras informaciones disponibles, para efectos de determinar la confiabilidad de la información.

El aspecto más importante de todo el sistema son los resultados, por lo que al revisar el sistema debe verificarse que los resultados cumplan con las especificaciones del diseño del sistema, lo cual debe comprobarse mediante juegos de datos de pruebas especialmente preparados, y que prueben todas las posibilidades de las entidades, datos y situaciones.

f. Efectividad del sistema

Uno de los aspectos más importantes del sistema, es que sea efectivo en conseguir los objetivos y beneficios esperados, por lo que se debe indagar a los usuarios sobre el grado de satisfacción y confiabilidad del sistema, que beneficios han sido conseguidos y los motivos que impiden lograr otros, si los costos de operación del sistema se encuentran dentro de lo planificado y que mejoras se han conseguido con tal reducción, que tanto ha mejorado la precisión de la información y en qué medida se ha completado y básicamente, cuanto se redujo el tiempo de atención al cliente y cuál fue el incremento de productividad para la institución poseedora del mismo.

g. Seguridad del sistema

Debe comprobarse que sistema cuente con los siguientes tipos de seguridad:

Seguridad de acceso a la información

Seguridad de acceso a los programas

Seguridad ante contaminación de virus

h. Sistema de respaldo

Previendo posibles problemas con el software es necesario que el equipo central cuente con características técnicas de respaldo como:

Listados diarios de información, lo cual permitiría seguir operando por lo menos manualmente en casos extremos.

Backup de la Base de Datos, deberían efectuarse por cada cambio de turno de trabajo o como mínimo al final del día, debiendo inclusive el sistema haber sido programado para que exija efectuarse el backup respectivo.

Copia de respaldo de los programas fuentes y objetos de las aplicaciones, de la plataforma de software y de las bases de datos completas de la Institución (hasta de tres períodos anteriores), debiéndose guardar una copia en el local de la Institución y otra adicional en otro local para afrontar siniestros o desastres que pudieran ocurrir.

i. Auditabilidad del sistema

En general, todo sistema que se encuentra instalado, debe contar con volúmenes de información como:

- Manual de usuario
- Manual de procedimientos de los sistemas
- Descripción genérica
- Diagramas de entrada, archivos, salida
- Salidas
- Fecha de instalación de los sistemas
- Proyecto de instalación de nuevos sistemas

4.2.2 Pruebas para auditar una aplicación Web

***Configuration Management Testing*:** "las pruebas descritas en esta categoría están orientadas a identificar fallos en las políticas de gestión de configuración.

Muchas veces los escaneos infraestructurales que puedan ser llevados a cabo revelarán información como puede ser, métodos HTTP permitidos, funciones administrativas y configuraciones infraestructurales"[31].

---

31 Metodologías OWASP http://www.seguridadparatodos.es/2013/04/OWASP-Parte3MetodologiaAppMobile.html [Citado el 16 de febrero de 2017]

Authentication Testing: En esta área se avalua todas las secciones de la web que estén relacionadas con los procesos de autenticación como puede ser el formulario de identificación, si es posible enumerar usuarios o si por ejemplo los mecanismos de captcha funcionan adecuadamente.

Session Management Testing: "Las técnicas descritas en esta sección se centrarán evaluando los controles de seguridad para las sesiones que se establecen controlando las interacciones de un usuario concreto, se mirará por ejemplo si substituyendo la sesión de un usuario por la de otro se podrá suplantar su identidad"[32] .

Authorization Testing: las pruebas englobadas en esta categoría van dirigidas a evaluar si los controles de autorización funcionan correctamente, por ello se mirará por ejemplo si los usuarios tienen acceso única y exclusivamente a los datos sobre los que están autorizados o si estos pudieran de alguna forma escalar privilegios.

Business Logic Testing: los fallos de lógica de negocio presentados en esta categoría son quizá los más complicados de identificar, pues requieren un profundo conocimiento de la aplicación y si existen son exclusivos para cada aplicación. Se trata de buscar aquellas funcionalidades en las cuales alterando el flujo normal del aplicativo puedan ser aprovechadas para beneficio del atacante. Se puede pensar en estos como aquellas funcionalidades que no cumplen correctamente las especificaciones del diseño, que no siguen los casos de uso.

4.2.3 Herramientas que existen actualmente para detección de vulnerabilidades en aplicaciones Web

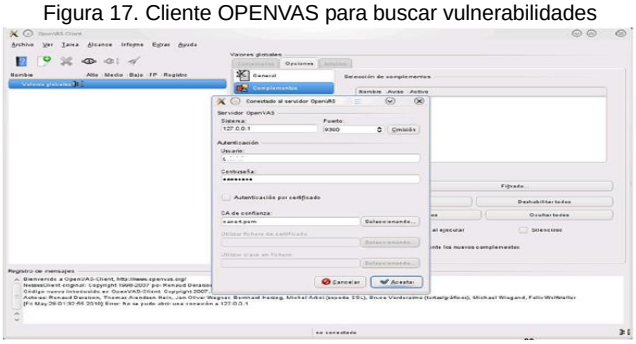
OPENVAS (*Open Vulnerability Assessment System*), es un *framework* de código abierto que analiza a profundidad que vulnerabilidades poseen los servicios que tiene instalado un Sistema Operativo, éste genera un reporte de utilidad que posteriormente se usa para parchear y corregir los problemas de seguridad de éstos.

---

32 Session Management https://cyberacademy.deloitte.es/cs-faqs/10-vulnerabilidades-de-proteccion-insuficiente-en-la-capa-de-transporte-2/ [Citado el 16 de febrero de 2017]

La siguiente gráfica muestra el cliente de Conexión de OpenVas, para iniciar sesión con el usuario y contraseña.

Figura 17. Cliente OPENVAS para buscar vulnerabilidades

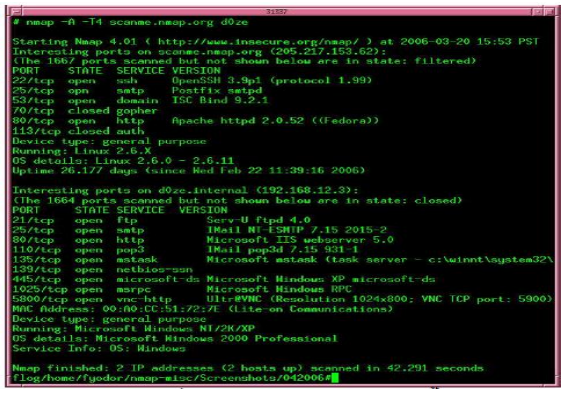


Fuente: http://labitacoradegJohan.wordpress.com/2016/03/08/instalando-openvas

NMAP, es una herramienta para escanear que servicios están disponibles y en que puertos, es muy útil para descubrir qué tipo de Sistema Operativo tiene el host analizado, como también que versiones de servicios y software tiene instalado.

En la siguiente imagen podemos observar un escaneo con ***nmap*** a los hosts scanme.nmap.org y d0ze, el parámetro -A es para habilitar la detección de Sistema Operativo y -T4 para una ejecución rápida.

Figura 18. **Escaneo de Servicios con NMAP**



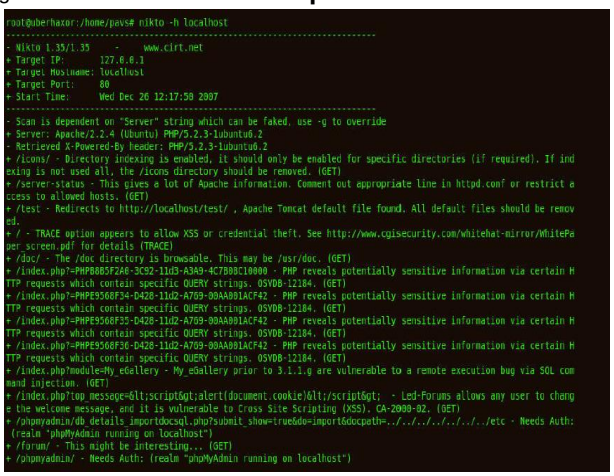
Fuente: http://nmap.org

NIKTO *"*Es un escáner de código abierto que lleva a cabo varias pruebas exhaustivas en los servidores web, incluyendo alrededor de 6400 archivos ***cgi*** potencialmente maliciosos, revisa además alrededor de 1200 versiones desactualizadas de servidores, y problemas específicos de más de 270 servidores. Éste también chequea la configuración y opciones del servidor, para

tratar de identificar que software trae instalado, como también la versión del servidor web”[33].

El parámetro -h específica el host o dominio objetivo como se observa en la siguiente gráfica:

Figura 19. **Escaneo de una aplicación Web con NIKTO**



Fuente: http://www.linuxhaxor.net/?p=648

SQLMAP “Es una herramienta de prueba de intrusión para automatizar el proceso de detección y explotación de fallas de inyección SQL, con el fin de tomar el control de la base de datos. Éste viene con un poderoso motor de detección que posee una amplia gama de pruebas, para acceder al sistema operativo y ejecutar consultas SQL”[34].

La herramienta se ejecuta de la siguiente manera:

python sqlmap.py -u http://www.justice.gov.al/index.php?gj=gj1--dbs

Donde -u es la URL o el enlace y --dbs es para enumerar las bases de datos disponibles.

---

[33] NC CIRT, Nikto [En línea], http://cirt.net/nikto2, [Citado el 14 de septiembre de 2011

[34] SQLMAP Developers, SQLMAP [En línea], <http://sqlmap.sourceforge.net/>, [Citado el 19 de septiembre de 2011]

Figura 20. **Ataque de Inyección SQL con SQLMAP** [35]

Fuente: https://es.wikipedia.org/wiki/Inyecci%C3%B3n_SQL

SQLNINJA **"**El principal objetivo de esta herramienta es explotar las vulnerabilidades de inyección SQL en aplicaciones Web que utilizan Microsoft SQL Server"[36].

Sqlninja se diferencia de otras herramientas, debido a que ellas están centradas a extraer información, en cambio Sqlninja se centra en conseguir una Shell interactiva con el servidor de bases de datos.

Para poder iniciar el ataque es necesario configurar el archivo sqlninja.conf que se encuentra en el mismo directorio de la herramienta, de la siguiente forma:

```
host = 172.16.24.151                # Host a atacar
port = 80                           # Puerto
method = GET                        # Método HTTP GET
page = /default.asp                 # Página principal del aplicativo
stringstart = id=1;                 # Variable GET y su valor de iniciación
stringend =
host = 172.16.24.150                # IP donde se ejecuta el análisis
msfpath = /opt/metasploit3/msf3/    # Path de metasploit Framework
```

Luego desde la consola ejecutamos:

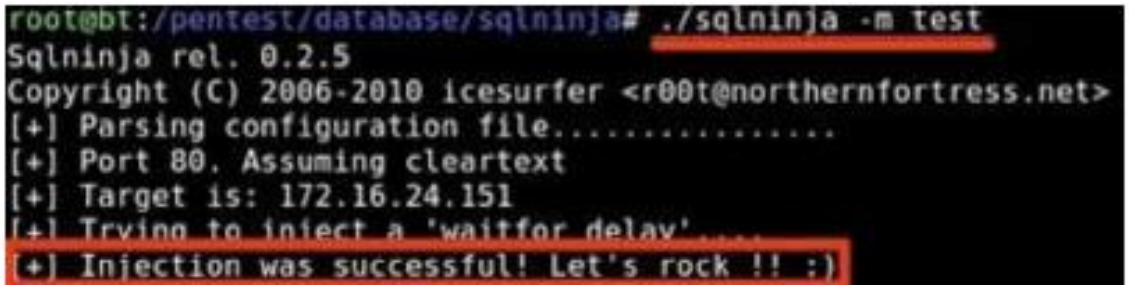
```
# ./sqlninja -m test
```

35 SQLMAP Developers, SQLMAP [En línea], <http://sqlmap.sourceforge.net/>, [Citado el 19 de septiembre de 2011]

36 ICESURFER, SQLNINJA [En línea], <http://sqlninja.sourceforge.net/sqlninja-howto.html#s1>, [Citado el 19 de septiembre de 2011]

En la siguiente gráfica se observa el resultado de la ejecución del comando:

Figura 21. Resultado del Ataque de Inyección SQL con SQLNINJA [37]

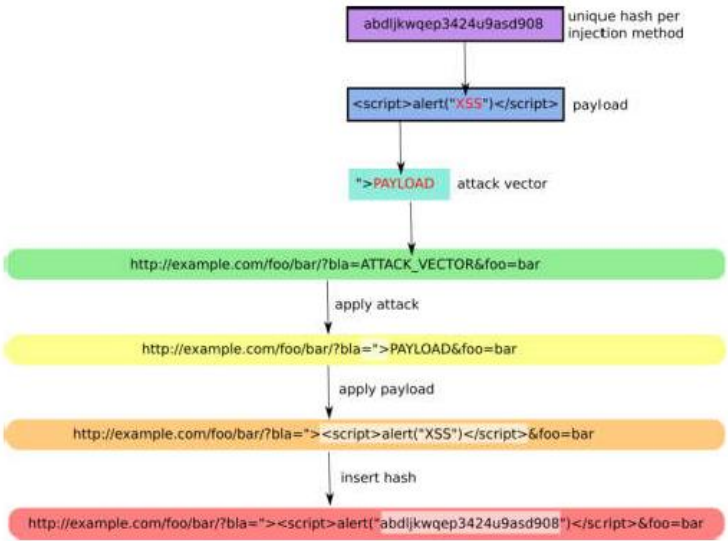


Fuente: El autor

XSSer "Cross Site "Scripter" es un framework que automatiza la detección y explotación, para reportar vulnerabilidades XSS en aplicaciones web. Además, posee varias opciones para tratar de eludir ciertos filtros y varias técnicas especiales que tratan de evitar la inyección de código"[38].

En la siguiente gráfica se explica el funcionamiento, donde a una variable por GETs, en este caso "bla" se inyecta código JavaScript para imprimir en una ventana un mensaje de texto:

Figura 22. Funcionamiento de ataque XSS con XSSER



Fuente: http://www.welivesecurity.com/la-es/2015/04/29/vulnerabilidad-xss-cross-site-scripting-sitios-web/

37 Selvi Savater, Jose, SQLNINJA [En línea], <http://www.pentester.es/2010/12/sql-injection-hastala-cocina-ms-sql.html>, [Citado el 20 de septiembre de 2011]

38 XSSer Workgroup, XSSer [En línea], <http://xsser.sourceforge.net/>, [Citado el 20 de septiembre de 2011]

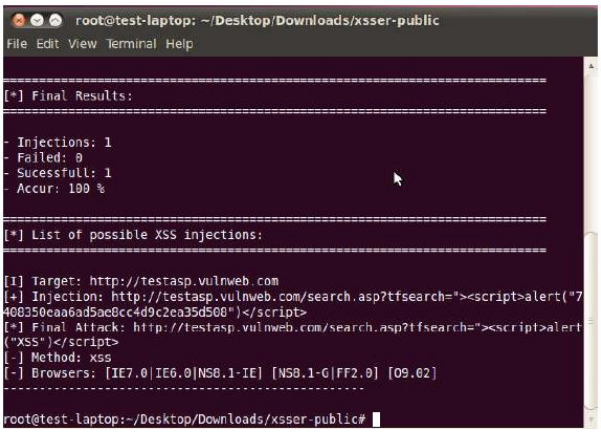
En esta otra gráfica se explica un ataque básico, donde python XSSer.py es el comando, -u es el dominio o host a atacar y -g especifica el archivo y su respectiva variable por GETs:

Figura 23. Parámetros necesarios para Iniciar XSSER

Fuente: El autor

El resultado de este comando se puede observar en la siguiente gráfica. En la cual se muestra una inyección de XSS exitosa y la URL del ataque.

Figura 24. Resultado del ataque de XSSER [39]



Fuente: https://xsser.03c8.net/

FIMAP Es una pequeña herramienta hecha en python que encuentra, prepara, audita y explota errores de inclusión local y remota de archivos en aplicaciones web.

En la imagen se puede observar el origen de la vulnerabilidad y el modo de ejecución de fimap. Donde -u es el host o dominio objetivo.

Una vez ejecutada la herramienta, se muestra que archivos son vulnerables a Remote File Inclusion (RFI).

---

39 Ethical Hacking-Your Way To The World Of IT Security, XSSer- Cross Site Scripting Penetration Tool [En línea], <http://www.ehacking.net/2011/02/xsser-cross-site-scripting-penetration.html>, [Citado el 20 de Septiembre de 2011]

Figura 25. **Escaneo de URL con Fimap** [40]

```
<?
  // Código PHP vulnerable a RFI
  include($_GET["inc"]);
?>
```

Explotando RFI con Fimap:

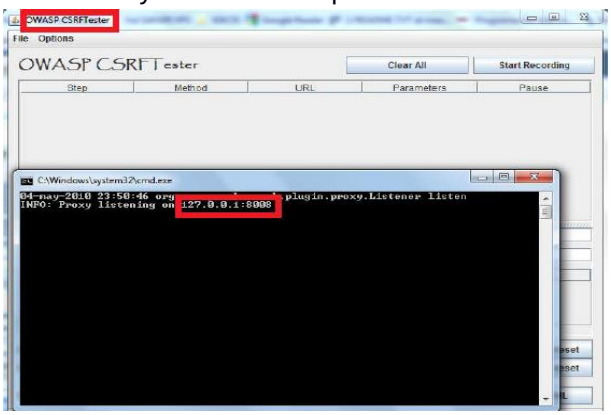
```
imax@DevelB0x:~$ fimap -u "http://localhost/vulnerable.php?inc=index.php
fimap v.01 by Iman Karim - Automatic LFI/RFI scanner and exploiter.
SingleScan is testing URL: 'http://localhost/vulnerable.php?inc=index.php
[OUT] Parsing URL 'http://localhost/vulnerable.php?inc=index.php'...
[INFO] Fiddling around with URL...
[OUT] Possible file inclusion found! -> 'http://localhost/vulnerable.php
[OUT] Identifing Vulnerability 'http://localhost/vulnerable.php?index.php
```

Fuente: http://calebbucker.blogspot.com.co/2012/06/fimap-automatic-lfirfi-scanner-and.html

CSRFTESTER "Es una herramienta de código abierto desarrollada en JAVA que actúa como un servidor proxy que depura las solicitudes HTTP en el navegador Web en busca de vulnerabilidades de tipo CSRF. Cabe resaltar que para utilizar esta aplicación se debe tener la máquina virtual de JAVA"[41].

Una vez ejecutada, ésta abre el puerto 8008 (Modo proxy). Además, el puerto debe configurarse en el navegador Web, donde se interceptarán todas las solicitudes HTTP en búsqueda de vulnerabilidades CSRF, para iniciar éste modo basta con dar click en "Start Recording", todo lo mencionado anteriormente puede observarse mejor en la siguiente imagen:

Figura 26. Interfaz y Consola de depuración de CSRFTESTER [42]



Fuente: http://www.adminso.es/recursos/Proyectos/PFM/2011_12/PFM_DVWA.pdf

---

[40] DALLA PIAZZA, Alessio, Fimap: Scanner LFI(Local File Inclusion) and RFI(Remote File Inclusion) [En línea], http://www.clshack.it/en/fimap-scanner-lfilocal-file-inclusion-and-rfiremote-fileinclusion.html, [Citado el 21 de Septiembre de 2011]

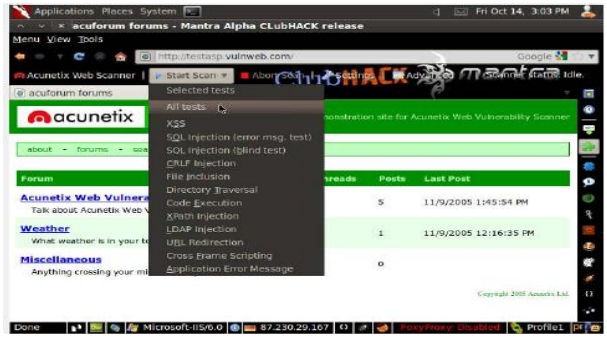
[41] OWASP FOUNDATION, CSRFTESTER [En línea], https://www.owasp.org/index.php/CSRFTester_Usage, [Citado el 21 de septiembre de 2011]

[42] Selvi Savater, Jose, Creando PoC CSRF: CSRFTester 1.0 [En línea], http://www.pentester.es/2010/05/creando-poc-csrf-csrftester-10.html, [Citado el 21 de septiembre de 2011]

OWASP MANTRA "Mantra es una colección de herramientas libres y de código abierto, integradas en un navegador basado en Firefox, la mayoría de estas herramientas son conocidas como complementos o extensiones del navegador"[43].

Lo primero para ejecutar estas herramientas es abrir la URL objetivo, en este caso http://testasp.vulnweb.com Luego de esto se da click en "Start Scan – All tests" como se observa en la siguiente imagen:

Figura 27. Prueba de Intrusión con Owasp Mantra



Fuente: https://www.owasp.org/images/8/80/Gu%C3%ADa_de_pruebas_de_OWASP_ver_3.0

El estado del escaneo puede observarse en "Scanner status:

Figura 1. Estado del Escaneo, Owasp Mantra

Fuente:
https://www.owasp.org/images/8/80/Gu%C3%ADa_de_pruebas_de_OWASP_ver_3.0

43 Owasp Mantra [En línea], http://www.getmantra.com/ [Citado el 16 de febrero de 2017]

WEBSECURIFY Es un complemento disponible para los navegadores Google Chrome y Firefox que utiliza un motor de reconocimiento de vulnerabilidades Web, también está disponible para versiones de escritorio en MAC, Windows y Linux.

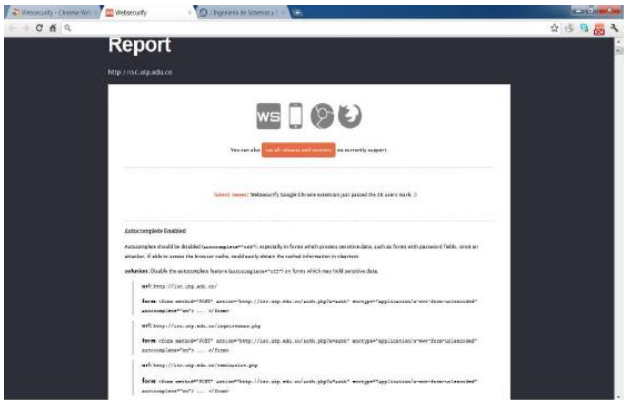
Para este caso se usará como una extensión del navegador. En la siguiente imagen se observa como inicia el proceso para el sitio http://isc.utp.edu.co:

Figura 28. Escaneo con Websecurify

Fuente: https://sourceforge.net/p/maguey/wiki/Websecurify-es/

El resultado del escaneo es mostrado como reporte de una manera muy interesante, dando una serie de recomendaciones:

Figura 29. Reporte Websecurify



Fuente: http://es.slideshare.net/Chely_princes/websecurify-an-dwebgoat-terminado

Kali Linux ha sido un sistema operativo que ha ganado terreno en el campo de la Seguridad informática.

Kali Linux es una distribución basada en Debian GNU/Linux diseñada principalmente para la auditoría y seguridad informática en general. Fue fundada y es mantenida por Offensive Security Ltd. Mati Aharoni y Devon Kearns, ambos pertenecientes al equipo de Offensive Security, desarrollaron la distribución a partir de la reescritura de BackTrack, que se podría denominar como la antecesora de Kali Linux.

Kali Linux trae preinstalados más de 600 programas incluyendo Nmap (un escáner de puertos), Wireshark (un sniffer), John the Ripper (un crackeador de passwords) y la suite Aircrack-ng (software para pruebas de seguridad en redes inalámbricas). Kali puede ser usado desde un Live CD, live-usb y también puede ser instalada como sistema operativo principal.

“Kali es desarrollado en un entorno seguro; el equipo de Kali está compuesto por un grupo pequeño de personas de confianza quienes son los que tienen permitido modificar paquetes e interactuar con los repositorios oficiales. Todos los paquetes de Kali están firmados por cada desarrollador que lo compiló y publicó”[44].

Figura 30. Herramientas Kali Linux

Fuente: https://es.wikipedia.org/wiki/Kali_Linux

44 Kali [Disponible en] https://www.offensive-security.com/kali-linux-vmware-virtualbox [Citado el 16 de febrero de 2017]

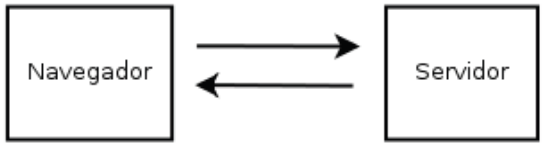
- **Herramientas para auditar una aplicación web**

Aunque existen herramientas genéricas para buscar de forma automática casi cada tipo de vulnerabilidad, éstas por lo general no suelen ser fiables. Los cambiantes entornos y las diferentes naturalezas de los sitios web hacen que éstas acaben requiriendo verificación manual.

Así que en cualquier caso se acabará requiriendo un análisis manual de vulnerabilidades.

Si así fuese se podría utilizar el navegador, como se ejemplifica en la figura **N° 16 Navegador conectado directamente al servidor remoto**, pero pronto se observará que éste queda pequeño ya que no proporciona control total sobre las peticiones y respuestas que envía y recibe de y hacia el servidor remoto. De igual modo utilizar un navegador para la evaluación manual puede resultar más complicado puesto que no está pensado para esta tarea.

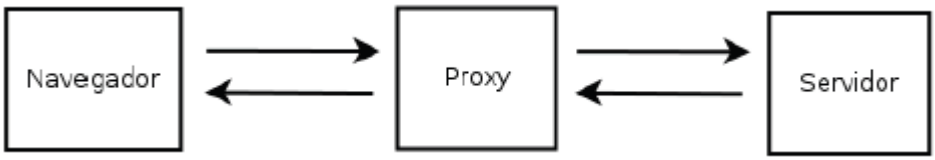
Figura 31. ***Navegador conectado directamente al servidor remoto***



Fuente: ***El autor***

Así que, con tal de realizar un análisis de vulnerabilidades, se necesitará otro tipo de aplicación que ofrezca más control sobre las peticiones y respuestas Es por ello que las herramientas que se presentan más útiles son las que se conocen como proxys de peticiones web o **proxys HTTP**.

Figura 32. ***Navegador conectado al servidor remoto mediante un proxy***



Fuente: ***El autor***

La mayoría de proxys de peticiones http permiten ver de forma sencilla e intuitiva el tráfico entre el navegador y el servidor puesto que se sitúan en mitad de la conexión y todo el tráfico es pasado a través de éstos como ha sido representado esquemáticamente en la figura anterior.

A continuación, se relaciona diferentes herramientas:

METASHIELD PROTECTOR: Módulo para IIS 7 capaz de eliminar los metadatos de los documentos ofimáticos.

De este modo con solo instalar este módulo todos los documentos accesibles públicamente a través de un portal no contendrán metadatos.

Es una herramienta centrada en buscar fugas de información de empresas a través de los metadatos, la información oculta y los datos perdidos que contenían los archivos públicos de una organización.

64POP3CONECTOR: Programa capaz de recoger mensajes de correo de buzones POP3 y enviarlos a un único Buzón. Esta herramienta es capaz de trabajar bajo conexiones seguras SSL/TLS. Las principales características del programa son:

- Ilimitadas Cuentas de Correo
- Ilimitadas Cuentas de Correo POP3
- Conexiones bajo SSL/TLS
- Fichero Log para seguimiento

SERVICIOS ONLINE: Foca Online, versión Online que permite extraer los metadatos de un documento.

4.3 PARTICULARIDADES DE LA EMPRESA C&M CONSULTORES

Haciendo el seguimiento técnico, monitoreo y control de la ejecución de contratos, la Empresa C&M CONSULTORES se ha convertido en la mano derecha de entidades públicas de diferentes sectores.

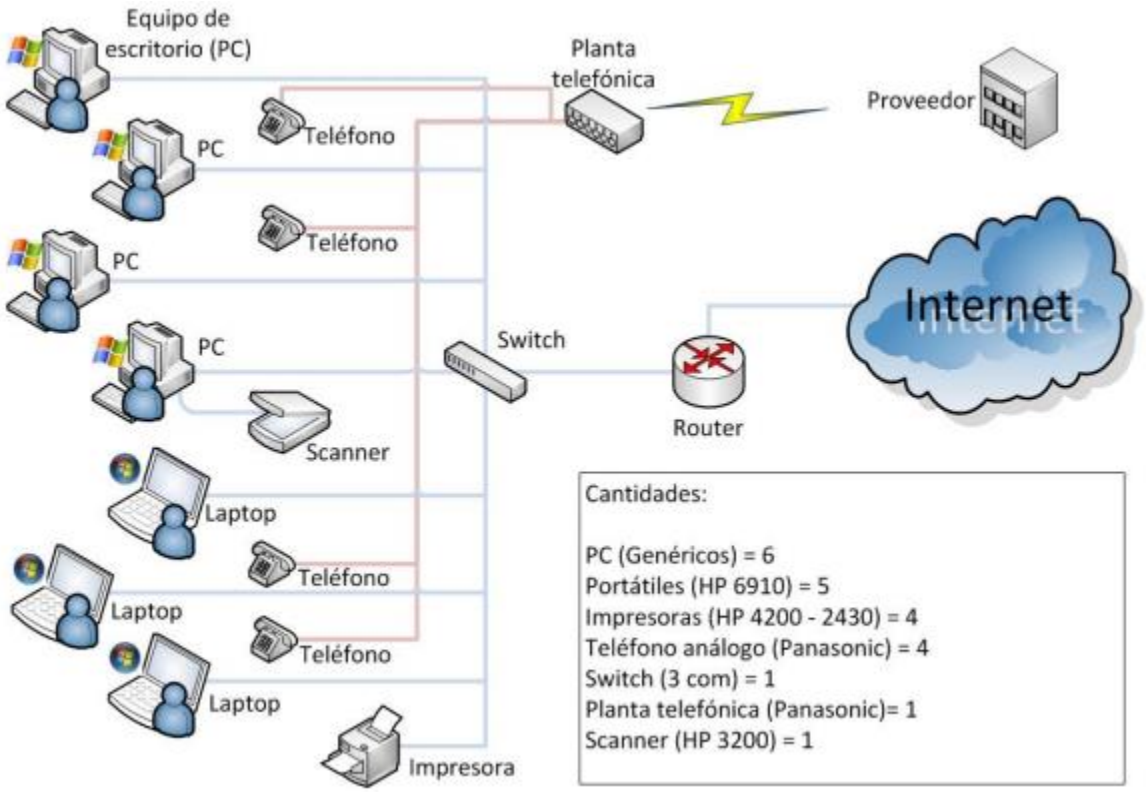
Debido a la interacción que los entornos web que realizan en la empresa, todas las aplicaciones y servicios web son potencialmente vulnerables a un gran número de ataques, independientemente de la plataforma y tecnología que utilicen. Estos ataques pueden producir desde deformación del sitio (cambio del contenido del sitio) o robo de datos hasta infección del sitio para la propagación de malware, lo que conlleva grandes pérdidas económicas sobre la organización afectada, su imagen y sus clientes.

La Auditoría Web o Auditoría de Aplicaciones Web tiene por objeto descubrir las vulnerabilidades de las aplicaciones o servicios web, detectando fallos de diseño

e implementación, falta de validación de entrada datos, etc. Para ello, se siguen metodologías de software libre.

En la figura 33 se puede observar e identificar diferentes equipos de cómputo de escritorio y otros equipos portátiles (Laptop), que dentro de la infraestructura se designan estaciones de trabajo. Todos ellos van interconectados a través de un switch de 24 puertos. Sobre este switch también se enlazan las respectivas impresoras y el router instalado por el proveedor de servicios de internet. Por otro lado, se tiene una planta telefónica que abastece servicio actualmente a 4 teléfonos tipo análogos. Los demás periféricos como scanner o cámaras se conectan de manera directa a las estaciones de trabajo. En este diagrama se puede observar claramente que todas las estaciones de trabajo tienen sistema operativo Windows, entre XP y Windows 7. Con la información anterior, se puede tener una concepción global de la infraestructura tecnológica.

Figura 33. Diagrama general de la infraestructura tecnológica



Fuente: Elaboración propia

Los recursos tecnológicos de forma general y cuantificada, sin entrar en detalle acerca de su configuración, estado o características particulares, todos los elementos son propios de C&M CONSULTORES, lo que genera autonomía sobre ellos. Todo el equipo de cómputo tiene sistema operativo Windows, ya sea

de la familia XP o 7. En cuanto al aspecto de la telefonía, la planta actualmente soporta tres líneas telefónicas y ocho extensiones de las cuales cuatro ya están siendo utilizadas. Estos teléfonos son análogos. Las impresoras y scanner se encuentran conectados directamente en los equipos de cómputo actualmente no están conectados en red, También se evidencia que el software antivirus que manejan para todos los equipos de cómputo es de tipo gratuito, y aunque puede ofrecer resultados positivos de protección es recomendable contemplar un antivirus de tipo pago el cual amplia aún más la protección necesaria.

C&M Consultores utiliza como su principal Aplicación Web, *SharePoint* con sus funciones de *Office* y servicios avanzados para mensajería, uso compartido de documentos, cumplimiento y características de administración para TI. Por lo tanto, no es tan complejo auditar esta aplicación Web, ya que se puede usar los siguientes informes de registro de auditoría proporcionados para ayudar a determinar quién realiza una acción determinada con el contenido de una colección de sitios:

- Revisión del contenido: Informa sobre los usuarios que han visto el contenido en un sitio.
- Modificaciones del contenido: Notifica los cambios que se realizan en el contenido, como modificar, eliminar o proteger y desproteger documentos.
- Eliminación: Notifica qué contenido ha sido eliminado.
- Tipo de contenido y modificaciones de lista: Notifica las adiciones, ediciones y eliminaciones en los tipos de contenido.
- Modificaciones de directiva: Notifica sobre los eventos que modifican las directivas de administración de información de la colección de sitios.
- Expiración y disposición: Notifica sobre todos los eventos relacionados con el modo en que se quita el contenido una vez que expira.
- Configuración de auditoría: Notifica sobre los cambios realizados en la configuración de auditoría.
- Configuración de seguridad: Notifica sobre los cambios realizados en la configuración de seguridad, como los eventos de usuario o grupo y los eventos de roles y derechos.
- Generar un informe personalizado: Puede especificar los filtros para un informe personalizado; por ejemplo, puede limitar el informe a un conjunto específico de eventos, a elementos en una lista en particular, a un intervalo de fechas determinado o a eventos realizados por usuarios concretos.

Así mismo, el correo corporativo es una dirección de correo electrónico que contiene el nombre comercial de la empresa. Se le llama corporativo porque así se diferencia de los correos electrónicos personales que por lo general son cuentas de correo gratis ofrecidas por los distintos servidores de correos en internet como lo son Gmail, Outlook, entre otros.

El correo corporativo a diferencia del correo electrónico gratis, mayormente tiene sistemas avanzados contra el spam y de protección contra virus. A menudo en la comunicación entre empresas es necesario bajar documentos adjuntos, los cuales pasan por un riguroso control antivirus antes de proceder a la descarga.

La interfaz webmail del correo corporativo tiene ventajas de cooperación, tales como:

- Calendario: se pueden crear eventos que se pueden compartir y a los que se pueden unir otros usuarios.
- Contactos: se pueden crear listas de contactos que se pueden compartir con otros usuarios.
- Tareas: se pueden crear tareas que se asignan a otros usuarios y se puede hacer seguimiento a las tareas.
- Compartir y editar documentos en la nube: se pueden compartir documentos y editarlos conjuntamente en la nube (solo algunos proveedores de correo ofrecen esta funcionalidad).

Actualmente se utiliza Office 365 (es una solución de arrendamiento del paquete Microsoft Office (Excel, Word, PowerPoint, Outlook y Access) por pagos mensuales en vez de pagar el producto completo por un precio elevado) como proveedor, tecnología Privada y documentos en la nube.

La arquitectura del software que administra las cuentas de correos corporativos tiene redundancias de seguridad para asegurar que no se pierdan mensajes ni se filtren virus.

Cada mensaje que se envía o recibe a través de un correo corporativo tiene una copia de seguridad.

Por otro lado, se cuenta con un sitio Web orientado a dar información sobre la empresa C&M CONSULTORES y defender la imagen en el mercado.

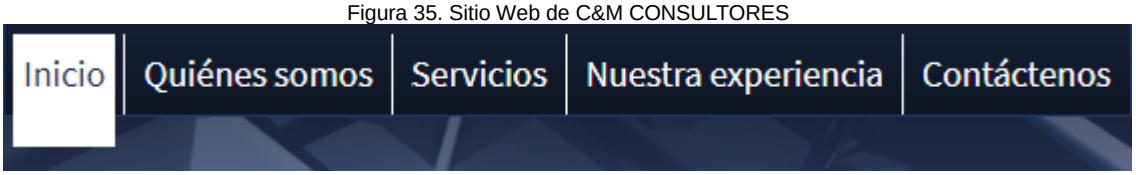
En la figura 34, se puede observar el sitio Web de la empresa.

Figura 34. Sitio Web de C&M CONSULTORES

Fuente: http://www.cmconsultores.com.co/

Como se puede ver en la figura 3, el sitio Web de la empresa cuenta con las siguientes secciones

Figura 35. Sitio Web de C&M CONSULTORES



Fuente: http://www.cmconsultores.com.co/

a. Inicio: Es la que se muestra cuando se abre el navegador por primera vez, detallando los datos de la empresa.

Figura 36. Sitio Web de C&M CONSULTORES

Fuente: http://www.cmconsultores.com.co/

b. Quienes somos: Detalla los antecedentes de la empresa y los clientes con los que ha interactuado.

Figura 37. Sitio Web de C&M CONSULTORES

Fuente: http://www.cmconsultores.com.co/

c. Servicios: Detalla los servicios que ofrece la empresa

Figura 38. Sitio Web de C&M CONSULTORES

Fuente: http://www.cmconsultores.com.co/

c. Nuestra experiencia: Relaciona los sectores en los que ha implementado sus servicios y algunas de las entidades a las que se ha asesorado.

Figura 39. Sitio Web de C&M CONSULTORES

Fuente: http://www.cmconsultores.com.co/

d. Contáctenos: Datos de la empresa para algún mensaje de su interés.

Figura 40. Sitio Web de C&M CONSULTORES

Fuente: http://www.cmconsultores.com.co/

Por último, cabe resaltar que la página Web es compatible con los navegadores más utilizados en los ordenadores u otros dispositivos móviles con acceso a la red. Su diseño se adapta a cualquier tamaño de pantalla y en estándares abiertos como HTML Y CSS.

4.4 PROPUESTA DE AUDITORÍA DE SEGURIDAD PARA APLICACIONES WEB DE C&M CONSULTORES

Resumen: La información es la entrada más importante para la empresa C&M CONSULTORES, asegurarla trae beneficios y credibilidad.

Es importante saber que hoy en día la información se encuentra en una aplicación, pero es más importante conocer que el avance de las tecnologías de la información ha llevado adquirir o desarrollar aplicaciones orientadas a la Web.

La propuesta está basada en metodologías existentes, para este proyecto será en software libre. Para tratar el tema de seguridad de la información, no se puede dejar a un lado ISO 27001, debido a que es el estándar para la seguridad en la información, que ofrece un modelo para el establecimiento, implementación, operación, monitorización, revisión, mantenimiento y mejoras de todos los procesos que involucran la seguridad de la información en una empresa por medio del Sistema de Gestión de la Seguridad de la Información.

Importancia de la seguridad de la información

Las buenas prácticas en el manejo adecuado de la seguridad de la información no solo disminuyen costos en la empresa, sino que también puede generar nuevas oportunidades de inversión.

La empresa C&M CONSULTORES tiene dentro de sus objetivos a corto plazo, mejorar de manera constante sus procesos, para esto gestionan y calculan cada parámetro, lo que les permite poder determinar cuándo una variación puede afectar la producción o los servicios que ofrecen.

Debido a que existen personas ajenas a la información, que buscan tener acceso a la empresa para modificar, sustraer o borrar datos. Surge esta pregunta ¿Por qué es tan importante la seguridad informática? Tales personas pueden, incluso, estar dentro de la empresa C&M CONSULTORES, de acuerdo a investigaciones realizadas para este proyecto, la mayoría de las violaciones e intrusiones a los recursos informáticos se realiza por el personal interno, debido

a que éste conoce los procesos, metodologías y tiene acceso a la información sensible de la empresa, es decir, a todos aquellos datos cuya pérdida puede afectar el buen funcionamiento de la organización.

Esta situación se presenta debido a los malos esquemas de seguridad con los que cuenta no solo la empresa C&M CONSULTORES sino la mayoría de las empresas a nivel mundial, y porque no existe conocimiento relacionado con la planeación de un esquema de seguridad eficiente que proteja los recursos informáticos de las actuales amenazas combinadas.

El resultado de todo esto, es la violación de los sistemas, provocando la pérdida o modificación de los datos sensibles de la organización, lo que puede representar un daño con valor de miles o millones de pesos.

Objetivo

Establecer una propuesta de auditoria de seguridad para aplicaciones Web de la empresa C&M CONSULTORES.

Plan de auditoría

La auditoría de una aplicación Web, debe ser objeto de una planificación cuidadosa. Es de crucial importancia acertar con el momento más adecuado para su realización:

- Por una parte, no conviene que coincida con el periodo de su implantación, en que los usuarios no dominan todavía la aplicación y están más agobiados con la tarea diaria.

- Por otra parte, el retraso excesivo en el comienzo de la auditoria puede alargar el periodo de exposición a riesgos superiores que pueden y deben ser aminorados como resultado de ella. Se recomiendan periodos entre 6 y 8 meses desde el inicio de la implantación.

- Hay que establecer el ámbito de actuación. se debe delimitar el campo de actuación de la mayor parte de las pruebas a realizar a un reducido número de centros de trabajo de campo.

- Debe conseguirse cuanto antes las autorizaciones necesarias para que el personal de auditoria, que está previsto participe en el trabajo, pueda

acceder a la aplicación y a las herramientas de usuario. Se solicitará como perfil de auditor, aquel que ofrezca las mayores posibilidades de consultas.

Propuesta Metodológica

La auditoría es el análisis examinador y sistemático que realiza a un individuo, organización, sistema, proceso, proyecto o producto. Existen varios tipos de auditorías y según su clasificación se ejecutan diferentes temas mediante una serie de métodos de investigación y análisis con el objetivo de producir la revisión y evaluación profunda de la gestión efectuada.

En la ejecución de la auditoría intervienen cuatro fases:

- Planeación
- Ejecución
- Informe
- Seguimiento

En la fase de Planeación se establece el provecho de los niveles de gestión óptimos en el proceso de auditoría. Es la fase donde se reúne información sobre la entidad auditada para determinar los riesgos y áreas de mayor importancia. Se definen los objetivos y alcances. Se elaboran los planes de trabajos generales e individuales. Se determinan los recursos humanos y materiales. Se selecciona la muestra a ser evaluada.

El objetivo de la etapa de la Ejecución de la Auditoría es obtener y analizar toda la información del proceso que se audita, con el propósito de obtener evidencia suficiente, competente y relevante, es decir, contar con todos los elementos que le aseguren al auditor el establecimiento de conclusiones fundadas en el informe acerca de las situaciones analizadas en el terreno. Durante esta fase se emiten los resultados parciales de la auditoría.

El Informe es el resultado de la investigación y análisis efectuados por los auditores durante la realización de la auditoría, que de forma normalizada expresa por escrito su opinión sobre el área o actividad auditada en relación con los objetivos fijados, señalan las debilidades de control interno, si las ha habido, y formula recomendaciones pertinentes para eliminar las causas de tales deficiencias y establecer las medidas correctoras adecuadas. Durante esta fase se informa el resultado final de la auditoría.

En la fase de Seguimiento el sujeto auditado presenta el plan de medidas, así como las medidas tomadas con los responsables en caso de existir hallazgos.

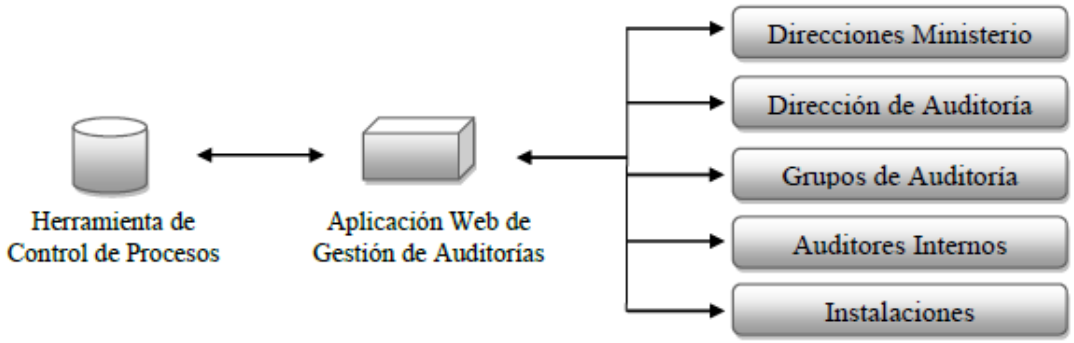
Durante el desarrollo de estas fases en la empresa analizada existe un constante flujo de información entre los implicados, La Dirección de Auditoría es la encargada de regir todo el proceso y rendir cuentas a la dirección de la empresa, tanto en aspectos de planificaciones, como en la toma de decisiones.

En el caso de los auditores internos, el proceso básicamente es similar, aunque solo ejecutan auditorías en las instalaciones donde laboran y están subordinados a la dirección de esta, pero mantienen una constante comunicación con los grupos de auditorías.

Resultados

Para dar solución al problema en cuestión se propone un sistema web que incorporará una herramienta de monitoreo y generación de avisos, como se muestra en la “Figura 3”. El sistema estará configurado para tener acceso desde todas las instancias y departamentos involucrados en el modelo de negocio.

Figura 2. Descripción de la Gestión de Auditoria



Fuente: http://www.captio.net/blog/5-herramientas-para-la-mejora-de-procesos

Identificación de vulnerabilidades

En esta fase se usan las herramientas de pruebas de intrusión

Las pruebas de vulnerabilidad permiten a la empresa tener una certeza razonable de hasta donde un ataque originado interna o externamente a la infraestructura de seguridad de la compañía puede ser efectivo y encontrar los

elementos a mejorar en dicha infraestructura constituyéndose así en una medida idónea para la revisión de la efectividad de los controles establecidos, poniéndolos a prueba frente a las sofisticadas herramientas para vulnerar su seguridad que están disponibles en Internet y que serán manejadas por un grupo de profesionales con amplia experiencia en el tema y un alto sentido de la ética.

- **Intrusión**

Una vez encontradas las vulnerabilidades, se descartan cuales son falsos positivos

- **Presentación de reporte**

Esta fase es primordial, debido a que con esta se demuestra que el sistema es seguro o no

Así mismo, una aplicación Web recorre las siguientes fases:

- Pre requisitos del Usuario (único o plural) y del entorno
- Análisis funcional
- Diseño
- Análisis orgánico (Preprogramación y Programación)
- Pruebas
- Entrega a Explotación y alta para el Proceso.

Estas fases deben estar sometidas a un exigente control interno, caso contrario, además del disparo de los costes, podrá producirse la insatisfacción del usuario.

Finalmente, la auditoria deberá comprobar la seguridad de las aplicaciones Web en el sentido de garantizar que los ejecutados por la maquina sean exactamente los previstos y no otros.

Una auditoria de aplicaciones pasa indefectiblemente por la observación y el análisis de cuatro consideraciones:

- Revisión de las metodologías utilizadas: Se analizarán éstas, de modo que se asegure la modularidad de las posibles futuras ampliaciones de la aplicación y el fácil mantenimiento de las mismas.

- Control Interno de las Aplicaciones: se deberán revisar las mismas fases que presuntamente han debido seguir el área correspondiente de Desarrollo:

- ❖ Estudio de Vialidad de la Aplicación. Importante para Aplicaciones largas, complejas y caras.
- ❖ Definición Lógica de la Aplicación. Se analizará que se han observado los postulados lógicos de actuación, en función de la metodología elegida y la finalidad que persigue el proyecto.
- ❖ Desarrollo Técnico de la Aplicación. Se verificará que éste es ordenado y correcto. Las herramientas técnicas utilizadas en los diversos programas deberán ser compatible.
- ❖ Diseño de Programas. Deberán poseer la máxima sencillez, modularidad y economía de recursos.
- ❖ Métodos de Pruebas. Se realizarán de acuerdo a las Normas de la Instalación. Se utilizarán juegos de ensayo de datos, sin que sea permisible el uso de datos reales.
- ❖ Documentación. Cumplirá la Normativa establecida en la Instalación, tanto la de Desarrollo como la de entrega de Aplicaciones a Explotación.
- ❖ Equipo de Programación. Deben fijarse las tareas de análisis puro, de programación y las intermedias. En Aplicaciones complejas se producirían variaciones en la composición del grupo, pero estos deberán estar previstos.

- Satisfacción de usuarios: Una Aplicación técnicamente eficiente y bien desarrollada, deberá considerarse fracasada si no sirve a los intereses del usuario que la solicitó. La aquiescencia del usuario proporciona grandes ventajas posteriores, ya que evitará reprogramaciones y disminuirá el mantenimiento de la Aplicación.

- Control de Procesos y Ejecuciones de Programas Críticos: El auditor no debe descartar la posibilidad de que se esté ejecutando un módulo que no se corresponde con el programa fuente que desarrolló, codificó y probó el área de Desarrollo de Aplicaciones. Se ha de comprobar la correspondencia biunívoca y exclusiva entre el programa codificado y su compilación. Si los programas fuente y los programa módulo no coincidieran se podría provocar, desde errores de bulto que producirían graves y altos costes de mantenimiento, hasta fraudes, pasando por acciones de sabotaje, espionaje industrial-informativo, etc. Por ende, hay

normas muy rígidas en cuanto a las Librerías de programas; aquellos programas fuente que hayan sido dados por bueno por Desarrollo, son entregados a Explotación con el fin de que éste:

- Copie el programa fuente en la Librería de Fuentes de Explotación, a la que nadie más tiene acceso
- Compile y monte ese programa, depositándolo en la Librería de Módulos de Explotación, a la que nadie más tiene acceso.
- Copie los programas fuente que les sean solicitados para modificarlos, arreglarlos, etc. en el lugar que se le indique. Cualquier cambio exigirá pasar nuevamente por el punto 1.

Ciertamente, hay que considerar las cotas de honestidad exigible a Explotación. Además de su presunción, la informática se ha dotado de herramientas de seguridad sofisticadas que permiten identificar la personalidad del que accede a las Librerías.

5 CONCLUSIONES

A continuación, se relacionan las principales conclusiones a las que se arribó:

- ✓ Se ha dado cumplimiento al objetivo de esta investigación, pues como resultado se obtuvo una propuesta de auditoria a las aplicaciones Web aplicando herramientas de Software libre, la cual podrá ser aplicada en otras instituciones.

- ✓ El costo de inversión es bajo ya que las tecnologías propuestas son libres, los resultados son profesionales y medibles a corto plazo.

- ✓ La seguridad en las aplicaciones debe considerarse desde el desarrollo. Debido a que reparar problemas de seguridad cuando la aplicación está terminada puede resultar muy costoso.

- ✓ Las Empresas ven la seguridad como un costo y no como un valor agregado que proporciona prestigio y confiabilidad con los clientes.

- ✓ Hablando de seguridad, es importante el aseguramiento del equipo y de las instalaciones, así como de la información, el control de los accesos también es punto muy importante para evitar las fugas de información o manipulación indebida de esta.

- ✓ El Departamento de informática es la parte medular de la empresa, es en donde los datos se convierten en información útil a las diferentes áreas, es donde se guarda esta información y por consecuencia, donde en la mayoría de los casos se toman las decisiones importantes para la empresa.

- ✓ Actualmente existen muchas herramientas que ayudan a detectar vulnerabilidades a través de pruebas de intrusión o revisiones de código que son gratuitas y libres, por lo tanto, es importante identificarlas para incluirlas en el proceso de desarrollo del software.

6 RECOMENDACIONES

Con base en el estudio desarrollado se ofrecen una serie de recomendaciones, las cuales ayudarán a mejorar a que la información de la empresa C&M CONSULTORES no sea vulnerable:

- Incluir dentro de la Política de Seguridad las políticas de Programación Segura, para mitigar las vulnerabilidades.
- Incluir dentro de los equipos existentes en el proceso de desarrollo al equipo de seguridad, quienes estarán presentes en cada una de las partes del proceso de desarrollo contribuyendo con los controles de seguridad.
- Incluir mecanismos de seguridad físicos tales como cortafuegos, IDS (Sistema de Detección de Intrusos), entre otros, para detectar posibles ataques al sistema y contrarrestarlos.
- Utilizar planes de contingencia para preservar el principio de disponibilidad, por si un sistema informático ha sido vulnerado.
- Si se identifican responsables de ataques informáticos al sistema, utilizar la normatividad 1273 de 2009 para proteger la organización y judicializar a los responsables.

REFERENCIAS

SCHNEIDER. PROYECTOS DE SEGURIDAD INFORMÁTICA. [Citado 24 de marzo, 2016]. Disponible en http://es.slideshare.net/raulosj/proyectos-de-seguridad-informtica

TARLOGIC. AUDITORÍA DE SEGURIDAD OWASP (AUDITORÍA WEB). [Citado febrero de 2015]. Disponible en: https://www.tarlogic.com/servicios/auditoria-de-seguridad-owasp/

Universidad de Alicante, Qué es una aplicación Web. [Citado el 5 de abril de 2016]. Disponible en: http://rua.ua.es/dspace/bitstream/10045/4412/5/03c-AplicacionesWeb.pdf

Wikispaces, Lenguajes de Programación. [Citado el 5 de septiembre de 2015]. Disponible en: http://cervantes1bachdyg.wikispaces.com/lenguajes_programacion

CODEBOX, Glosario [En línea], [Citado el 5 de febrero de 20161]. Disponible en: http://www.codebox.es/glosario

PORTAL HACKER, Programación en General. [Citado el 6 de marzo de 2016]. Disponible en: http://www.portalhacker.net/index.php/topic,115175.0/wap2.html

New Web Star, Los diferentes lenguajes de programación para la web. [Citado el 6 de marzo de 2016]. Disponible en: http://www.newwebstar.com/ebooks/133193-los-diferentes-lenguajes-deprogramaciun-para-la.html

Ingeniería de Requerimientos, Arquitectura de Software. [Citado el 7 de mayo de 2016]. Disponible en: http://proypnfi.foroactivo.net/search.forum?search_author

ALDEGANI, Gustavo. Miguel. Seguridad Informática. MP EDICIONES. ARGENTINA. 1997. Pág. 22 CATEGORY: OWASP INSECURE WEB APP PROJECT/ES (Citado el 22 de marzo 2015) Disponible en: https://www.owasp.org/index.php/Category:OWASP_Insecure_Web_App_Project

CATEGORY: OWASP PANTERA WEB ASSESSMENT STUDIO PROJECT/ES. (Citado octubre 2015) Disponible en: https://www.owasp.org/index.php/Category:OWASP_Pantera_Web_Assessment_Studio_Project/es

SCHNEIDER, Ben. Outsourcing: La herramienta de gestión que revoluciona el mundo de los negocios, Grupo EDITORIAL NORMA, Pág. 183. AGUILERA, Purificación. Seguridad Informática, EDITEX, Pág. 22

LÓPEZ BROX, Antonio. Promociones en espacios comerciales, EDITORIAL VERTICE, Pág. 395

SM Stuart. Hackers, Secretos y soluciones para seguridad de redes, Osborne-McGrawHill, 2015

KLUS, Como auditar controles de aplicación [Citado el 31 de enero de 2016]. Disponible en: https://www.auditool.org/blog/auditoria-de-ti/265-na-breve-introduccion-de-como-auditar-controles-de-aplicacion

FEDESOFT, Colombia es el cuarto país más vulnerable de América Latina en seguridad informática [Citado el 16 de abril de 2016]. http://www.fedesoft.org/noticiastic/colombia-el-cuarto-pais-mas-vulnerable-de-america-latina-en-seguridad-informatica

Stuart McClure, SM, Hackers, Secretos y soluciones para seguridad de redes, Osborne-McGrawHill, 2001

New Web Star, Los diferentes lenguajes de programación para la web [En línea], [Citado el 6 de septiembre de 2011]

Ingeniería de Requerimientos, Arquitectura de Software [En línea], [Citado el 7 de septiembre de 2011]

ALDEGANI, Gustavo. Miguel. Seguridad Informática. MP EDICIONES. ARGENTINA. 1997. Pág. 22

SOFTTRON NET, Seguridad Informática [En línea], [Citado el 7 de septiembre de 2011]

ASENSIO, Gonzalo. Seguridad en Internet, EDITORIAL NOWTILUS, Pág. 15.

AGUILERA, Purificación. Seguridad Informática, EDITORIAL EDITEX, Pág. 11. 7 de septiembre de 2011] [28] AGUILERA, Purificación. Seguridad Informática, EDITORIAL EDITEX, Pág. 12-27.

ANEXOS

Anexo 1 RESUMEN RAE

RESUMEN ANALITICO ESPECIALIZADO - RAE	
1. Título.	PROPUESTA DE UNA AUDITORIA A LAS APLICACIONES WEB DE LA EMPRESA C&M CONSULTORES APLICANDO HERRAMIENTAS DE SOFTWARE LIBRE
2. Autor:	Johan Lorenzo Contreras Flórez
4. Año de elaboración	2017
5. Palabras Claves,	Vulnerabilidad, aplicaciones web, herramientas, auditoria, monitoreo, metodología, Software libre, seguridad
6. Descripción.	En el desarrollo del proyecto, en primer lugar, se especifica toda la estructura y funcionalidad de las vulnerabilidades en las aplicaciones web, basado en software libre, para establecer una auditoria del proyecto. En segundo lugar, se determinan los tipos de vulnerabilidades en las aplicaciones web y las herramientas que existen en la actualidad para la detección de vulnerabilidades en aplicaciones web. En tercer lugar, se realiza un monitoreo de las actividades en la red y los recursos informáticos de la empresa C&M CONSULTORES.
7. Fuentes.	1. PROYECTOS DE SEGURIDAD INFORMÁTICA (Consultado octubre 2015) Disponible en http://es.slideshare.net/raulosj/proyectos-de-seguridad-informtica 2. TARLOGIC. AUDITORÍA DE SEGURIDAD OWASP (AUDITORÍA WEB). (Consultado octubre 2015) Disponible en: https://www.tarlogic.com/servicios/auditoria-de-seguridad-owasp/ 3. CATEGORY: OWASP INSECURE WEB APP PROJECT/ES (Consultado octubre 2015) Disponible en: https://www.owasp.org/index.php/Category:OWASP_Insecure_Web_App_Project/es 4. CATEGORY: OWASP PANTERA WEB ASSESSMENT STUDIO PROJECT/ES. (Consultado octubre 2015) Disponible en: https://www.owasp.org/index.php/Category:OWASP_Pantera_Web_Assessment_Studio_Project/es 5. SCHNEIDER, Ben. Outsourcing: La herramienta de gestión que revoluciona el mundo de los negocios, Grupo EDITORIAL NORMA, Pág. 183

	6. AGUILERA, Purificación. Seguridad Informática, EDITEX, Pág. 22 7. LÓPEZ BROX, Antonio. Promociones en espacios comerciales, EDITORIAL VERTICE, Pág. 395 8. SM Stuart. Hackers, Secretos y soluciones para seguridad de redes, Osborne-McGrawHill, 2001 9. WORDLINGO, Prueba de concepto [En línea], <http://www.worldlingo.com/ma/enwiki/es/Proof_of_concept#In_security>, [Citado el 31 de agosto de 2011] 10. FEDESOFT, Colombia es el cuarto país más vulnerable de América Latina en seguridad informática [En línea], <http://www.fedesoft.org/noticiastic/colombia-el-cuarto-pais-mas-vulnerable-de-america-latina-en-seguridad-informatica>, [Citado el 31 de agosto de 2011] 11. Stuart McClure, SM, Hackers, Secretos y soluciones para seguridad de redes, Osborne-McGrawHill, 2001
8. Contenidos.	INTRODUCCIÓN 1. TITULO 2. FORMULACIÓN DEL PROBLEMA 3. JUSTIFICACIÓN 4. OBJETIVOS 5. MARCO REFERENCIAL 6. DISEÑO METODOLOGICO 7. RESULTADOS E IMPACTOS 8. DIVULGACIÓN
9. Metodología.	Las herramientas de Software libre, son metodologías que tiene en cuenta la seguridad en el ciclo de desarrollo del Software. Muchas referencias, actividades y experiencias en la temática de aplicaciones web, están apoyadas en este proyecto, como un Ingeniero de Sistemas, estudiante de la especialización en seguridad informática con las intenciones de plantear este proyecto, que a su vez trabajará en temas de seguridad de aplicaciones, buenas prácticas para desarrollo seguro, pruebas de seguridad para software entre otros
10. Conclusiones.	1. Mantener el Sistema Operativo donde está la

	aplicación Web, debido a que puede ser una puerta de entrada. 2. Incluir dentro de la Política de Seguridad las políticas de Programación Segura, para mitigar las vulnerabilidades. 3. Incluir dentro de los equipos existentes en el proceso de desarrollo al equipo de seguridad, quienes estarán presentes en cada una de las partes del proceso de desarrollo contribuyendo con los controles de seguridad. 4. Incluir mecanismos de seguridad físicos tales como cortafuegos, IDS (Sistema de Detección de Intrusos), entre otros, para detectar posibles ataques al sistema y contrarrestarlos. 5. Utilizar planes de contingencia para preservar el principio de disponibilidad, por si un sistema informático ha sido vulnerado. 6. Si se identifican responsables de ataques informáticos al sistema, utilizar la normatividad 1273 de 2009 para proteger la organización y judicializar a los responsables.
11. Autor del RAE.	Johan Lorenzo Contreras Florez