Premio Universitario
ESET 2015

Seguridad en Base de Datos & Aplicaciones Web

LUIS ARTURO FELICIANO CARDONA
CARNÉ: 0903-10-12483

UNIVERSIDAD MARIANO GALVEZ | GUATEMALA C.A.

1 CONTENIDO

2 INTRODUCCIÓN

Iniciamos hablando sobre LA CALIDAD. Según la NORMA INTERNACIONAL ISO 9000, la calidad es el "grado en el que un conjunto de características inherentes cumple con los requisitos". De forma sencilla podemos entender a la calidad en aplicaciones web como la satisfacción de las necesidades de los usuarios, que sean confiables y sobre todo que cuenten con la mejora continua en su desempeño.

Hoy día vivimos en una serie de transiciones tecnológicas que cada vez hacen de nuestras actividades cotidianas tales como compras, ventas, transacciones monetarias, etc., un proceso sencillo, confiable, optimizando los recursos más importantes como el tiempo y costo. Con la web semántica, el proceso será aún mejor ya que será posible intercambiar información entre dos o más sistemas informáticos, haciendo uso de los agentes inteligentes.

Sin duda las aplicaciones web son cada vez más utilizadas a nivel mundial. Sin embargo esto nos lleva a la siguiente interrogante: ¿Son seguras las nuevas aplicaciones web? De forma rápida describimos la estructura de una aplicaciones web distribuida en capas: Capa 1: Aplicaciones cliente. Capa 2: Servidor Wen. Capa 3: Servidor de base de datos. Siendo la última capa lo más importante de cualquier sistema. Nuestro activo más importante se resume en la siguiente palabra: Información.

.

3 SUMMER

We begin by talking about QUALITY. According to the INTERNATIONAL STANDARD ISO 9000, quality is the "degree in which a group of inherent characteristics fulfills the requirements." Simply speaking we can understand quality in web applications as satisfying the users' needs, that they be reliable and especially that they have continual improvement in performance.

Today we live in a series of technological transitions that increasingly make our everyday activities such as buying, selling, monetary transactions, etc., a simple, reliable process, optimizing the most important resources such as time and cost. With the Semantic Web, the process will be even better since it will be possible to exchange information between two or more computer systems, using intelligent agents.

No doubt web applications are increasingly used worldwide. But this brings us to the question: are the new web applications secure? Briefly we describe the structure of a web application distributed in layers: Layer 1: Client applications. Layer 2: Web Server. Layer 3: Server database. The latter being the most important layer of any system. Our most important asset is summarized in the following word: information.

"No se puede asegurar lo que no se conoce"

Principios básicos de seguridad informática

4 MALWARE

El "Malware" es uno de los términos que más hacemos uso, a la hora de referirnos a la seguridad informática. En sí ¿Qué es Malware? Veamos lo que nos dice Eldad Eilam: "El software malicioso (o malware) es cualquier programa que va en contra del propietario o del usuario del sistema. En términos generales, los usuarios esperan que el equipo y que todo el software que se ejecuta en él trabaje para su beneficio. Cualquier programa que viole esta regla se considera malware, porque funciona para el interés de otras personas." [1]

El malware es una palabra compuesta de "malicious" y "software" es decir software o código malicioso. Como lo vimos anteriormente, es todo aquel software que perjudica el funcionamiento de una computadora. Pero éste no sólo perjudica el funcionamiento, sino puede ir más allá, al tratar de una o varias formas el cómo aprovechar recursos informáticos y sobre todo obtener información sensible de usuarios, organizaciones, empresas, bancos hasta incluso información clasificada de gobiernos. Existe una gran cantidad de malware que se diferencian por la forma de propagación, los daños que ocasionan entre otras características. Veamos el siguiente resumen de la clasificación de Malware según ESET[2]:

Virus	Produce algún daño en el sistema del usuario. Posee 3 características particulares: infecta archivos, actúa de forma transparente al usuario y tiene la capacidad de reproducirse a sí mismo.
Gusanos	Su característica no es infectar archivos sino la modificación de claves en el Registro para ejecutarse en el inicio del sistema o explotar vulnerabilidades en el OS o aplicaciones. Es capara de reproducirse a sí mismo. Medios utilizados según la variante: Envío por Email, cliente IM, propagación por redes P2P, por vulnerabilidades de software, por dispositivos USB.
Troyanos	Simula ser inofensivo, útil o benigno para el usuario. No infecta archivos y necesita del usuario para su propagarción. Se caracterizan por una alta utilización de técnicas de Ingeniería Social. Existe una gran cantidad de troyanos los cuáles suelen clasificarse según el daño que causan, entre ellos están los siguientes: Downloader, Banker, Dropper, Clicker, Keylogger, Backdoor y los Bot.
Adware	Se instala en el sistema sin que el usuario lo note, cuya función es descargar y/o mostrar textos o imágenes de publicidad en la pantalla de la víctima. El adware utiliza información recopilada por algún spyware.
Spyware	Conocido como progrma espía, es una aplicación cuyo fin es recolectar información del usuario sin su consentimiento. La información puede ser: Historial de navegación, Descarga de archivos, Compras vía e-commerce, Información demográfica (edad, sexo, etc.), Intereses comerciales.

[1] Reversing Secrets of Reverse Engineering (Pág. 273)
[2] ESET Análisis de malware – Conceptos básicos (Págs. 7-14)

Rogue	Simula ser un programa de seguridad, con el fin de lograr que el usuario pague por una aplicación dañina o inexistente. Intenta generar miedo en el operador de la PC, indicando falsas alertas sobre infecciones y/o problemas que pudiera tener el sistema. Utilizado por 2 fines: Instalación de malware y cobro de dinero.
Ransomware	Cifra o bloquea el acceso a la información; para que el usuario pueda volver acceder a ella necesita realizar el pago de una determinada cantidad de dinero. Algunas de las acciones que se incluyen en su funcionamiento son: Cifrado de archivos de disco, bloque total de acceso al sistema, bloqueo de ciertos archivos.
Rootkit	Herramienta diseñada para mantener en forma encubierta el control de una computadora. Es decir cualquier componente lógico que permita al atacante mantener el acceso y el control del sistema: programas, archivos, procesos, puertos, etc. El rootkit no es un software maligno en sí mismo, sino que permite ocultar las acciones dañinas que ejecute un atacante en el sistema.

La ingeniería social es otro tema muy importante en cuanto a la seguridad informática. De acuerdo a ***Christopher Hadnagy*** se define a la Ingeniería Social como "El acto de manipular a una persona para que lleve a cabo una acción que -puede ser o no- lo más conveniente para cumplir con cierto objetivo. Este puede ser la obtención de información, conseguir algún tipo de acceso o logar que se realice una determinada acción".[3]

Existen personas con altos grados de conocimientos en informática, en donde hacen uso de sus conocimientos con el objetivo de delitos informáticos. Los laboratorios de ESET a nivel mundial identifican aproximadamente 240,000 nuevos códigos maliciosos por día. Es increíble la cantidad de malware en donde el usuario final está expuesto. Para su propagación existe diversidad de metodologías y técnicas. En esta oportunidad simplemente mencionamos los más conocidos: Las redes sociales, los correos electrónicos en donde se hace uso de los (spam, hoax, scam), el Phishing.

A nivel regional, las empresas encuestadas en Colombia, Venezuela, Ecuador, Perú y Nicaragua son la que reportaron mayores niveles de infecciones con códigos maliciosos.[4]

Esto no es una sorpresa cuando observamos que, por ejemplo, en Colombia las detecciones de botnets crecieron un 10% durante 2014, y

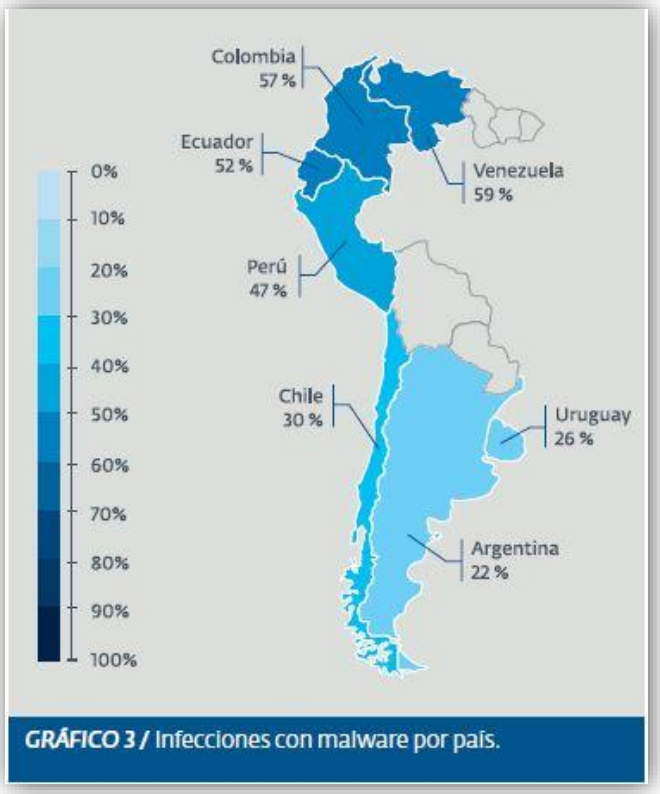


GRÁFICO 3 / Infecciones con malware por país.

[3] Ingeniería Social. El Arte del Hacking Personal, Christopher Hadnagy.
[4] ESET Security Reports Latinoamérica 20015

en países como Ecuador y Venezuela encontramos variantes de códigos maliciosos que no son los más comunes comparados con el resto de Latinoamérica.[5]

5 BASE DE DATOS

5.1 CONTROL DE ACCESO.

El control de acceso se define en dos enfoques: Control de acceso direccional y el control de acceso obligatorio. El control de acceso direccional es cuando un usuario específico tendrá generalmente diferentes derechos de acceso (también conocidos como privilegios) sobre diferentes objetos.[6] Es decir; cuando un administrador de base de datos otorga los privilegios a cada usuario, éste puede tener diferentes tipos de acceso según las tareas que se le asignen. El DBA garantiza privilegios a usuarios, incluyendo la capacidad para acceder archivos de datos específicos, registros o campos para operar de una manera determinada (read, insert, delete, o update).

Ejemplo:

```
GRANT SCHEMA Nombre EsqBD AUTHORIZATION usuario;
GRANT privilegios ON objeto TO usuarios [WITH GRANT OPTION]
REVOKE [GRANT OPTION FOR] privilegio ON objeto FROM usuarios
{CASCADE | RESTRICT}
```

El control de acceso obligatorio es cuando cada objeto de datos está etiquetado con un nivel de clasificación determinado y a cada usuario se le da un nivel de acreditación y los esquemas obligatorios tienden a ser jerárquicos por naturaleza.[6] Lo que quiere decir, es que el control de acceso obligatorio cede los privilegios de forma jerárquica de tal manera que la seguridad de toda la organización esté dividida estratégicamente y así obtener un mejor control en cuanto quién puede tener acceso a información confidencial.

Este nivel de seguridad en cuanto a los controles de acceso, obligan a que un objeto de datos específico sólo puede ser accedido por los usuarios que tengan el nivel de acreditación adecuado y principalmente clasifica los usuarios y datos en múltiples niveles de seguridad, y luego fuerza determinadas reglas acordes a cada nivel.

5.2 CONTROL DE ACCESO BASADO EN ROLES:

El acceso obligatorio o mandatorio es rígido porque hay que asignar permisos de una determinada clase a cada objeto y sujeto.[6] En el mundo real, los privilegios de acceso están asociados con el rol de una persona en la organización.

- Cada rol debe ser creado con determinados privilegios.
- Cada usuario es asociado a un rol.

[5] Botnets a la orden del día. http://www.welivesecurity.com/la-es/2014/10/23/zombis-colombia-ecuador-venezuela-botnets/
[6] Jorge Sánchez Asenjo (2009) Apuntes Completos Sistemas Gestores de Base de Datos. España

Este tipo de control, según las necesidades, puede ser el más adecuado, ya que según los roles que cada usuario desempeña, así será su tipo de privilegio, en cuanto más delicado sea el rol, más delicado será su responsabilidad con la información de la compañía.

5.3 CONTROL DE ACCESO MEDIANTE TRIGGERS

Con la utilización de los triggers es posible crear mecanismo de seguridad más complejos que pueden ser disparados cada vez que se llama un evento.[7]

El comando `Insert` en la tabla es un ejemplo de un evento que puede ser usado para disparar un Triggers, además de eso, los mismos pueden ser disparados antes o después del comando especificado con el objetivo de proveer mayor rigor en el control de la seguridad. Si el comando ejecutado por el usuario no es validado por los Triggers, salta un error en el cuerpo del propio trigger para impedir que la tabla sea modificada indebidamente.

Los triggers la utilizamos generalmente para asegurar reglas de negocio complejas y auditar cambios en los datos, es por ello la necesidad inmediata de los controles.

5.4 CONTROL DE ACCESO MEDIANTE VIEWS

Las vistas constituyen otro método de control de acceso, normalmente son utilizadas para restringir el acceso directo a los datos.[7]

Con la view es posible permitir el acceso de un usuario concediendo privilegios, ocultar líneas y columnas de informaciones confidenciales o restringir a los residentes en la tabla original de las indicaciones del SQL. Los privilegios y concesiones están definidos solamente en la view y no afectan a la tabla base, estando el acceso de los usuarios delimitando por la view, la cual se genera creando un subconjunto de datos en la tabla referenciada.

Privilegio sobre una vista: Para el caso de las vistas podemos a un usuario otorgar permisos de la siguiente forma. Como Ejemplo hacemos uso de la tabla alumnos y el campo Luis.

```
SELECT, INSERT, UPDATE, DELETE, DEBUG, REFERENCES.
SQL > GRANT ALL ON vista_alumnos TO Luis
```

Otorgamos al usuario Luis todos los permisos sobre la vista vista_alumnos.

5.5 GRANT – REVOKE

Un usuario o papel al que se le concede un privilegio no está autorizado de manera predeterminada a concedérselo a otros usuarios o papeles. Si se desea conceder un privilegio a un usuario y permitirle que lo transmita a otros usuarios hay que añadir la cláusula with grant option a la orden grant correspondiente.[8]

[7] Gabriel Gallardo Avilés (Seguridad en Base de Datos y Aplicaciones Web)

[8] Fundamentos de Base de Datos. Cuarta Edición. Madrid. Mc Graw Hill

A un usuario podemos otorgarle una serie de privilegios. Un privilegio permite a un usuario acceder a ciertos objetos o realizar ciertas acciones:

- Privilegios sobre Objetos (Object privileges)
- Privilegios del Sistema (System privileges)
- Privilegios sobre Roles (Role privileges)

Para otorgar privilegios utilizamos la sentencia GRANT, para quitar un privilegio o permiso a un usuario utilizamos la sentencia REVOKE.[9]

Ejemplo:

Privilegio sobre una tabla:

```
SQL > GRANT ALL ON tabla_alumnos TO byron
```

Siendo tabla_alumnos una tabla de nuestra base de datos y byron un usuario de esta, hemos asignado mediante `GRANT ALL`, todos los permisos al usuario byron sobre esta tabla.

```
GRANT ALL = permisos SELECT, INSERT, UPDATE, DELETE
```

Si queremos asignar sólo uno de estos permisos utilizamos la misma sentencia pero con el permiso que queramos otorgar.

```
SQL > GRANT SELECT ON tabla_alumnos TO byron
SQL > GRANT SELECT, INSERT ON tabla_alumnos TO byron
```

QUITANDO PRIVILEGIOS

Si queremos quitar un privilegio a uno de estos objetos haremos lo mismo que con GRANT pero utilizando la sentencia `REVOKE.`

```
SQL > REVOKE ALL ON tabla_usuarios FROM byron
```

5.6 ENCRIPTACIÓN DE PASSWORD.

Nunca debemos guardar las contraseñas de los usuarios sin cifrar. Esto es un error más común de lo que pueda parecer ya que los clientes suelen pedir como requisito el que se pueda "recordar" la contraseña a los usuarios (enviar por correo la contraseña actual, no una nueva).[10]

Las contraseñas siempre han de guardarse en la base de datos "cifradas" de algún modo, de forma que el atacante no pueda conocerlas. Si es requisito indispensable el que las contraseñas puedan ser "recordadas", deberán cifrarse con un algoritmo de "doble sentido" para que puedan ser descifradas por la aplicación. Pero este sistema es muy

[9] Jorge Sánchez Asenjo (2009) Apuntes Completos Sistemas Gestores de Base de Datos. España

[10] Luis M. (s/f) Encriptar y Guardar Contraseñas en Base de Datos.

poco recomendable ya que el atacante podrá también descifrarlas. Lo mejor es cifrar las contraseñas con un algoritmo de "un sólo sentido" de forma que no se puedan descifrar.

6 OLAP

Según la Wikipedia, OLAP es: "OLAP es el acrónimo en inglés de procesamiento analítico en línea (On-Line Analytical Processing). Es una solución utilizada en el campo de la llamada inteligencia empresarial (o Business Intelligence) cuyo objetivo es agilizar la consulta de grandes cantidades de datos. Para ello utiliza estructuras multidimensionales (o cubos OLAP) que contienen datos resumidos de grandes bases de datos o Sistemas Transaccionales (OLTP). Se usa en informes de negocios de ventas, marketing, informes de dirección, minería de datos y áreas similares."[11]

Veamos el siguiente ejemplo de un comercial de Pizza Freschetta, en donde vemos como una mujer haciendo una combinación de datos y filtrando características de una pizza entre varias. Finalmente encuentra la deseada.

Porque OLAP no es más que: Una manera de acceder a tu información utilizando un "lenguaje natural".

En el mundo corporativo, los analistas acceden a la información filtrando sus indicadores de negocio por regiones, por producto, por tiempo, etc. A partir de esta definición básica, existen distintas tecnologías que lo implementan (ROLAP, MOLAP), pero básicamente todas hacen las mismas acciones básicas sobre la información:

- Segmentar: Como cuando pides las ventas por producto y por trimestre.
- Filtrar: Como cuando pides el informe de ventas de Toyota en San Marcos, Guatemala.

[11] WIKIPEDIA recuperado en https://es.wikipedia.org/wiki/OLAP

- Profundizar (Drill down): Como cuando ves los datos de trimestre 2 y te interesa el desglose de abril, mayo, junio.
- Sintetizar (Drill up): Cuando deshaces el desglose anterior y vuelves al desglose por trimestre.
- Rotar (Drill anywhere): Cuando en lugar de pasar de un desglose por trimestres a uno mensual, te interesa un desglose por familia de producto, o por nacionalidad, es decir, por una característica de una jerarquía distinta a la que lo estás viendo actualmente.[12]

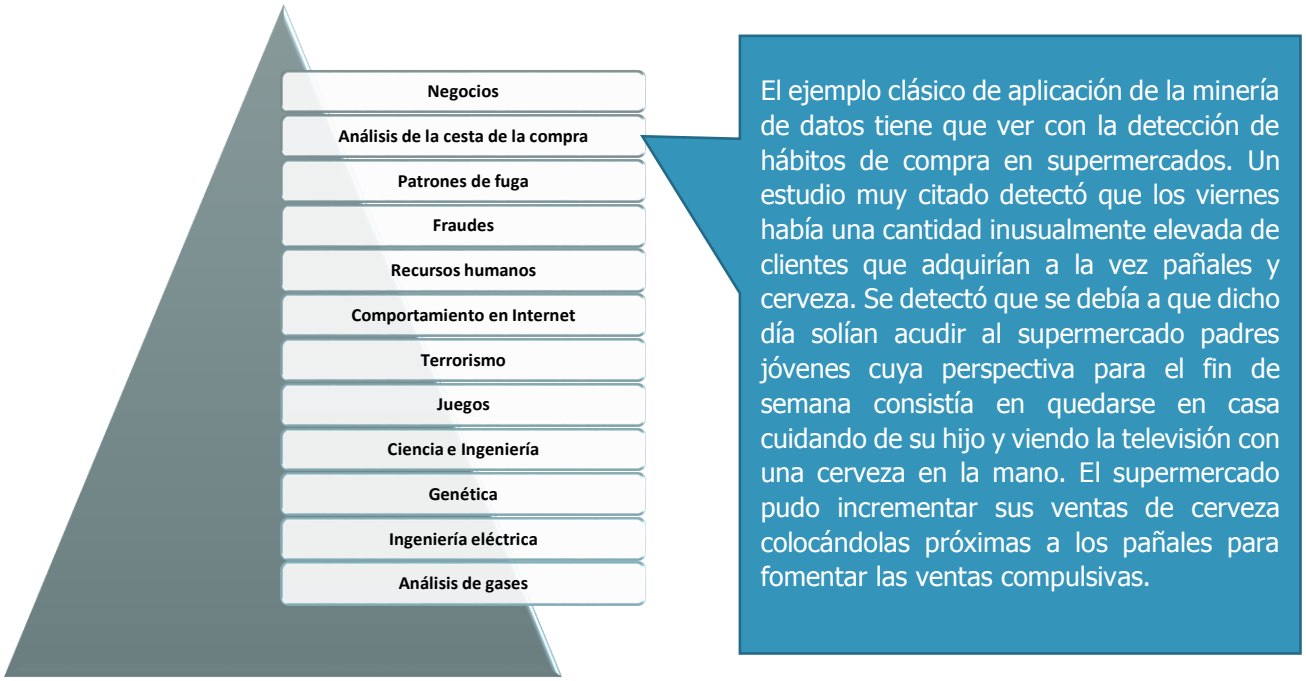
OLAP nos permite "navegar" fácilmente por la información, solicitándola con el detalle preciso y con los filtros adecuados, y que podemos hacerlo de manera dinámica, fácil, ad hoc, sobre la marcha, sin necesitar asistencia, rápido, y utilizando el lenguaje de negocio.

7 MINERÍA DE DATOS.

La minería de datos es el proceso de detectar la información accionale de grandes conjuntos de datos. Utiliza el análisis matemático para deducir los patrones y tendencias que existen en los datos. Normalmente, estos patrones no se pueden detectar mediante la exploración tradicional de los datos porque las relaciones son demasiado complejas o porque hay demasiado datos.[13]

Todas las herramientas tradicionales de minería de datos asumen que los datos que usarán para construir los modelos contienen la información necesaria para lograr el propósito buscado: obtener suficiente conocimiento que pueda ser aplicado al negocio (o problema) para obtener un beneficio (o solución).

Ejemplos de uso de la minería de datos



[12] Pau Urquizu, socio de Crono Analytics. Business Intelligence recuperado en fácilhttp://www.businessintelligence.info/definiciones/que-es-olap.html
[13] Microsoft – Recuperado en: https://msdn.microsoft.com/es-es/library/ms174949(v=sql.120).aspx

8 LA INTERCEPCIÓN DE DATOS CONFIDENCIALES

Según las investigaciones que se han realizado para escribir la presente monografía, es interesante dar a conocer la postura de **Marc, J** (s/f):

El cuarto riesgo importante que amenaza a la empresa es la intercepción por parte de un tercero de datos confidenciales. Es necesario tomar conciencia de un hecho: es una conexión de red y/o internet normal, el 99.9% de los datos que circulan no están cifrados por lo que pueden ser interceptados por cualquiera. Es más, es una operación simple y al alcance de cualquier pirata. Existen muchos programas que permiten guardar y luego consultar todo lo que pasa por una red informática, llamados "Packet sniffer"(p. 23)

Tal como se expresa en la cita anterior, la confidencialidad de los datos de una compañía representa gran valor; por lo que se hace necesario los conocimientos necesarios en cuanto a los pasos a seguir para proteger la información de cualquier hacker o cracker en la red.

9 APLICACIONES WEB.

Una aplicación web es una página web especial, que tiene información sobre la que se puede interactuar e incluso cambiar. La diferencia con las aplicaciones de escritorio es que no se instala ni se ejecuta en tu ordenador, sino a través de un navegador. Ejemplos de aplicaciones web son: Gmail, Hotmail, Google.

Muchas aplicaciones web tienen descuidos en programación y de administración. Si estos sitios están vulnerables, de algún modo exponen información o datos. Como una de las metas del hacking ético es encontrar esas brechas antes de que un atacante real lo haga, en nuestra tarea como profesionales éticos la clave es buscar, interpretar, analizar, generar errores y revisar de modo intensivo, hasta encontrar. No todo lo encontrado puede ser explotable, pero podría ser una muy buena pista para dar con algo sensible, o bien, mejorar la seguridad.[14]

10 INYECCIÓN DE CÓDIGO SQL

Es impresionante el alcance que tiene un descuido en los activos por la deficiente implementación de seguridad informática, tanto en el código como en la administración de un contexto donde se lleva a cabo la gestión de información con base de datos relacionales. (*Tori, 2008, 166*); pareciera que los efectos que puede ocasionar la deficiente administración de la seguridad informática, específicamente la seguridad en las bases de datos, no afectaran grandemente, pero desafortunadamente grandes compañías han perdido gran cantidad de información a causa de esta problemática.

"El descuido que permite inyectar código SQL (Structured Query Language) va a perdurar tanto, por lo humano, y es tan interesante". (*Tori, 2008, 166*), actualmente, un gran número de sitios y aplicaciones web interactúan con bases de datos, teniendo en cuenta que a través de éstos, se maneja información de diferentes niveles de criticidad, que deben ser almacenados, consultados y modificados, es decir, gestionados; toda esta información es accedida a través de sentencias SQL, embebidas desde el mismo código fuente de la página o scripts incluidos.

[14] Hacking Ético por Carlos Tori 2008

11 BYPASS DE ACCESO

En términos generales. Cuando nos referimos al bypass de acceso nos referimos a un salto de login. Es decir tratamos de evadir el ingreso a un sitio web.

Veamos el siguiente ejemplo para comprender mejor una vulnerabilidad. En el código, se verá la sintaxis transaccional SQL a la base de datos: [15]

```
SELECT id
FROM login
WHERE usuario = ‘$usuario’
AND clave = ‘$clave’
AND cuenta = ‘$cuenta’

SELECT id
FROM login
WHERE usuario = ‘admin’
AND clave = ’ OR 1=1
```

Algunos ejemplos de inyección para bypass de accesos son:

- “ or 0=0 #
- ‘ or 1=1 –
- a” or “a”=”a
-)’or 1=1;
- or 0=0 #
- “ or “a”=”a

12 CROSS-SITE SCRIPTING

El Cross-Site-Scripting, conocido también como XSS por sus siglas en inglés, es una vulnerabilidad que muchos desarrolladores dejan pasar por alto, quizás por falta de un planeamiento de análisis de riesgos en el proceso de diseño, desarrollo e implementación de sus aplicativos, o simplemente no lo vean como una falla que les va a presentar problemas en su aplicación, incluso por desconocimiento de esta vulnerabilidad.

[15] Hacking Ético por Carlos Tori 2008

Sea cual sea el motivo por el cual no se ejecuten medidas de prevención con respecto a esta vulnerabilidad, es necesario saber, que es un tipo de ataque que se extiende cada día más, hoy en día es muy común encontrar sitios y usuarios afectados por este tipo de agresión.

Para ejecutar este tipo de ataque no es necesario ser un gurú en el campo de la programación, ni desarrollar o utilizar herramientas complejas, basta nada más con manejar un poco de las etiquetas de HTML y algún lenguaje de Scripting, es suficiente para utilizar el cross-site-scripting en sitios que no están protegidos contra este tipo de ataque.

Ahora bien, la forma de Proteccion también son métodos básicos y de fácil implementación, que no conllevan ninguna tarea complicada que signifique un retrazo significativo en el tiempo de desarrollo de la aplicación, ni un quebradero de cabeza para implementar en sitios ya desarrollados, ya que dicha solución consiste en implantar un sistema de validación en los campos de entrada de datos, además de tener claro que no podemos fiarnos de datos que vengan de nuestras propias bases de datos cuando estos datos pueden ser insertados o modificados por el usuario.

La clave primordial en el campo de la seguridad es no confiar en la entrada de datos de ningún usuario, ya que en algún momento estas entradas se puede convertir en una inserción de código malicioso, es por ello que la aplicación debe de validar cada uno de los campos en donde el usuario pueda o le sea requerido el ingreso de datos y dejar de lado la política de confianza al usuario.

13 METADATOS

Comúnmente, se define a los metadatos como un conjunto de datos sobre datos. Si lo llevamos a la vida diaria, un ejemplo podría ser el siguiente: si el dato en cuestión es un libro, la ficha que podríamos tener sobre ese libro en una biblioteca serían los metadatos, es decir, su autor, fecha de publicación, editorial y demás especificaciones del libro (dato). [16]

Para el caso de archivos como fotos, música y documentos de ofimática, estos archivos también traen consigo metadatos que, en muchas ocasiones, servirán para buscar un archivo creado en una fecha específica, de un autor preciso e inclusive saber con qué calidad se encuentra un archivo de audio.

[16] ESET - Guía de Privacidad en Internet
PUBLICADO 28 SEP 2015 - http://www.welivesecurity.com/wp-content/uploads/2015/09/eset-guia-privacidad-internet.pdf

Sin embargo, hay veces en las cuales a través de imágenes se puede conocer una posición geográfica (mediante coordenadas GPS) en el caso de los Smartphone, o subiendo algún archivo de ofimática a la nube se puede ver el nombre de usuario de un equipo. Es por esto que se debe tener un cuidado especial entendiendo que la información que se sube a Internet puede contener (o brindar) más datos que meramente lo que se ve en una foto o se muestra en un archivo.

Si bien en la actualidad la mayoría de las Redes Sociales elimina los metadatos, no se puede saber a ciencia cierta si futuras redes también lo harán, por lo cual nos parece importante resaltar su existencia y los posibles peligros que traen aparejados

Ejemplo:

¿Una foto habla más de la cuenta?

Si no se toman los recaudos necesarios, sí: esta simple foto podría entregar información muy sensible a un atacante. A continuación veremos qué información trae escondida nuestro "amiguito" amarillo.

14 RECOMENDACIONES

14.1 EVALUACIÓN DE LA VULNERABILIDAD Y LA CONFIGURACIÓN

Evalúe su configuración de bases de datos, para asegurarse que no tiene huecos de seguridad.

Esto incluye la verificación de la forma en que se instaló la base de datos y su sistema operativo (por ejemplo, la comprobación privilegios de grupos de archivo -lectura, escritura y ejecución- de base de datos y bitácoras de transacciones). Asimismo con archivos con parámetros de configuración y programas ejecutables. Además, es necesario verificar que no se está ejecutando la base de datos con versiones que incluyen vulnerabilidades conocidas; así como impedir consultas SQL desde las aplicaciones o capa de usuarios. Para ello se pueden considerar (Como administrador):

- Limitar el acceso a los procedimientos a ciertos usuarios.
- Delimitar el acceso a los datos para ciertos usuarios, procedimientos y/o datos.
- Declinar la coincidencia de horarios entre usuarios que coincidan.

14.2 ENDURECIMIENTO (HARDENING EN DB)

Como resultado de una evaluación de la vulnerabilidad a menudo se dan una serie de recomendaciones específicas. Este es el primer paso en el endurecimiento de la base de datos. Otros elementos de endurecimiento implican la eliminación de todas las funciones y opciones que se no utilicen. Aplique una

política estricta sobre que se puede y que no se puede hacer, pero asegúrese de desactivar lo que no necesita.[17]

14.3 AUDITE

Una vez que haya creado una configuración y controles de endurecimiento, realice auto evaluaciones y seguimiento a las recomendaciones de auditoría para asegurar que no se desvíe de su objetivo (la seguridad).

Automatice el control de la configuración de tal forma que se registre cualquier cambio en la misma implementando una bitácora. Implemente alertas sobre cambios en la configuración. Cada vez que un cambio se realice, este podría afectar a la seguridad de la base de datos.

14.4 MONITOREO

Monitoreo en tiempo real de la actividad de base de datos es clave para limitar su exposición, aplique o adquiera agentes inteligentes de monitoreo, detección de intrusiones y uso indebido.

Por ejemplo, alertas sobre patrones inusuales de acceso, que podrían indicar la presencia de un ataque de inyección SQL, cambios no autorizados a los datos, cambios en privilegios de las cuentas, y los cambios de configuración que se ejecutan a mediante de comandos de SQL.

Recuerde que el monitoreo usuarios privilegiados, es requisito para la gobernabilidad de datos y cumplimiento de regulaciones como SOX y regulaciones de privacidad. También, ayuda a detectar intrusiones, ya que muchos de los ataques más comunes se hacen con privilegios de usuario de alto nivel.

El monitoreo dinámico es también un elemento esencial de la evaluación de vulnerabilidad, le permite ir más allá de evaluaciones estáticas o forenses. Un ejemplo clásico lo vemos cuando múltiples usuarios comparten credenciales con privilegios o un número excesivo de inicios de sesión de base de datos.

14.5 TESTEO WEB

El auge de las aplicaciones Web en el mercado de software ha propiciado el desarrollo de sistemas utilizando metodologías convencionales, sin considerar los nuevos problemas que este tipo de soluciones traen aparejadas.

El comercio electrónico, las aplicaciones distribuidas, el trabajo colaborativo, entre otras, son actividades comunes en el desarrollo de software convencionales y muy factibles de migrar a aplicaciones basadas en la Web. Pero si

[17] Gabriel Gallardo Avilés (Seguridad en Base de Datos y Aplicaciones Web)

esta tarea es realizada de forma directa, sin tener en cuenta las diferencias existentes, no tendrá un éxito. Si se considera un sistema de escritorio tradicional y se migra a un entorno Web.[18]

- **Clientes dinámicos:** Ya no se tiene un cliente de forma estática, sino que el servidor va a ir generando dinámicamente las interfaces del cliente para cada petición.
- **Clientes desde Sistemas Operativos diferentes:** Es posible que los clientes se conecten desde sistemas operativos diferentes, por lo que las interfaces deben funcionar correctamente en cada uno de ellos.
- **Diferentes tipos de conexión:** Los clientes pueden conectarse a través de diferentes tipos de redes. Los puntos de conexión pueden variar de velocidades, tiempo de respuesta, protocolos de transmisión, etc.
- **Alteraciones del control de flujo por parte del usuario:** En una aplicación convencional los usuarios no pueden alterar el flujo del programa, mientras que en las aplicaciones Web los usuarios pueden alterar el flujo de control presionando las teclas de retroceso o actualización de página, cambiando el contexto de ejecución y generando efectos inesperados. Estas posibilidades deben estar contempladas en el desarrollo de una aplicación Web.
- **Cambios de configuración:** El usuario puede cambiar la configuración del cliente (como por ejemplo deshabilitando cookies), produciendo así un cambio de comportamiento en la aplicación.
- **Problemas en la programación:** Existen características particulares en el desarrollo de sistemas orientados a la web, que generan una necesidad de mayor testeo que en las aplicaciones de escritorio estándar. Algunas de esas características son la utilización de varios lenguajes para su codificación (entre otros, HTML o Javascript en el cliente y Java o Perl para ser ejecutado en el servidor), el alto re-uso de código existente (sin implicar que este código ya esté testeado) y la utilización de componentes de terceros (que no siempre son garantía de calidad).
- **Problemas de interacción:** Los sistemas distribuidos basados en la Web se caracterizan por utilizar o brindar servicios de otros sitios. Tal es el caso de los sistemas que admiten pagos con tarjetas de crédito, en los cuales se interactúa con un sistema bancario que brinde efectivamente el servicio de cobro con tarjeta.
    - La información que utiliza un sistema Web puede estar distribuida en diferentes bases de datos ubicadas en diversos puntos geográficos.
    - Además, la usabilidad característica de estos sistemas, hace que personas con diversas aptitudes sean usuarios potenciales.
    - Las particularidades enunciadas generan un problema inherente en la interacción de los sistemas web con otros sistemas, con las bases de datos manipuladas y con la gran diversidad de usuarios hipotéticos existentes, incrementándose la necesidad de testeo de estas aplicaciones.
- **Mantenimiento:** Las tecnologías utilizadas en las aplicaciones basadas en la Web avanzan muy rápido, lo que requiere que el mantenimiento se deba llevar a cabo más frecuentemente y de forma más eficiente, especialmente en lo que respecta a seguridad.

15 CONCLUSIÓN

Sin duda la importancia de la seguridad en aplicaciones web y bases de datos son una prioridad alta en toda compañía. Como vimos que un grupo de crackers exponen 38 millones de infidelidades.

El grupo de crackers The Impact Team ha accedido a las bases de datos que albergaban en Avid Life Media. Vicente Fernández, director del Grupo SAI y Luis Fernando García Alcaraz, experto en seguridad informática del citado grupo. Se les hace la siguiente pregunta: ¿Qué técnicas actuales se conocen que permitan hackear una web y robar

[18] Offutt J., "Quality Attributes of Web Software Applications".

los datos de los usuarios, como el caso de Ashley Madison? Respuesta: Una de las más conocidas es SQL Injection, que trata de actuar de manera activa sobre la base de datos desde la propia página WEB víctima.

Para evitar cualquier robo de información, es necesario implementar una serie de políticas de seguridad a distintos niveles. Evaluación de riesgos sin duda es indispensable realizar. Un constante monitoreo y test de seguridad periódicas tanto a las bases de datos como a las aplicaciones web.

16 REFERENCIAS

- Para imágenes utilizadas: http://www.freeimages.com
- Seguridad de Base de Datos: http://www.teamshatter.com/
- Offutt J., "Quality Attributes of Web Software Applications". IEEE Software: Special, Issue on Software Engineering of Internet Software 19 (2):25-32, Marzo/ Abril 2002.
- ESET Latinoamérica. Análisis de malware (Conceptos básicos) Juan Díaz de Solís 1270, 2do. Pisco – Vicente López. Buenos Aires.
- Autor: Jorge Sánchez Asenjo (2009) Apuntes Completos Sistemas Gestores de Base de Datos. España
- Abraham Silberschatz, Henry F. Korth y S. Sudarshan. (2002) Fundamentos de Base de Datos. Cuarta Edición. Madrid. Mc Graw Hill
- Gabriel Gallardo Avilés (Seguridad en Base de Datos y Aplicaciones Web).
- Kenneth C. Laudon Jane P. Laudorn (2008) Sistemas de Información Gerencial (Administración de la Empresa Digital) Décima Edición. México. Pearson Educación
- Luis M. (s/f) Encriptar y Guardar Contraseñas en Base de Datos. Recuperado de: http://www.arumeinformatica.es/blog/encriptar-y-guardar-contrasenas-en-base-de-datos/
- Microsoft (2015) Recuperado de: https://technet.microsoft.com/es-es/library/ms365343(v=sql.105).aspx
- Reversing Secrets of Reverse Engineering (Pág. 273)
- Análisis de malware – Conceptos básicos (Págs. 7-14)
- Ingeniería Social. El Arte del Hacking Personal, Christopher Hadnagy.
- ESET Security Reports Latinoamérica 20015
- WIKIPEDIA recuperado en https://es.wikipedia.org/wiki/OLAP.
- Hacking Ético por Carlos Tori 2008

17 ANEXOS

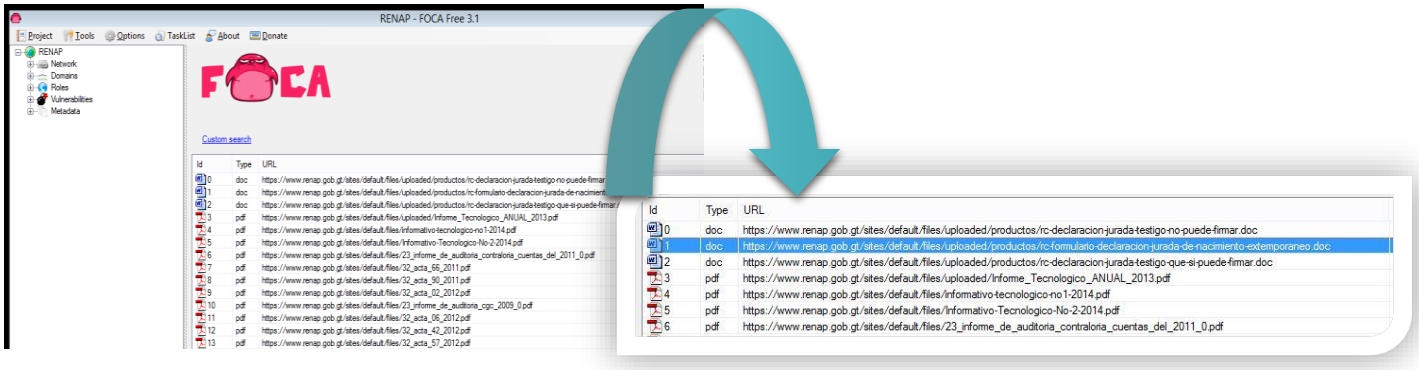
SEGURIDAD DEL SITIO WEB DE RENAP

Seguridad de Sitio Web : https://www.renap.gob.gt/

Herramienta Utilizada: FOCA Free 3.1

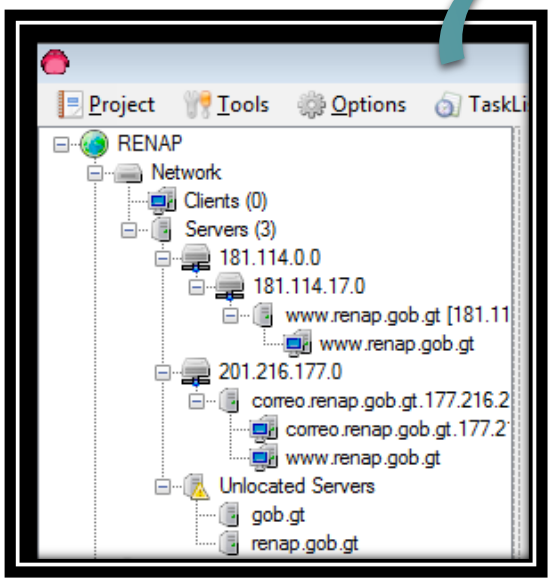
FOCA Free es una herramienta para la realización de procesos de fingerprinting e information gathering en trabajos de auditoría web. La versión Free realiza búsqueda de servidores, dominios, URLs y documentos publicados, así como el descubrimiento de versiones de software en servidores y clientes. FOCA se hizo famosa por la extracción de metadatos en documentos públicos, pero hoy en día es mucho más que eso

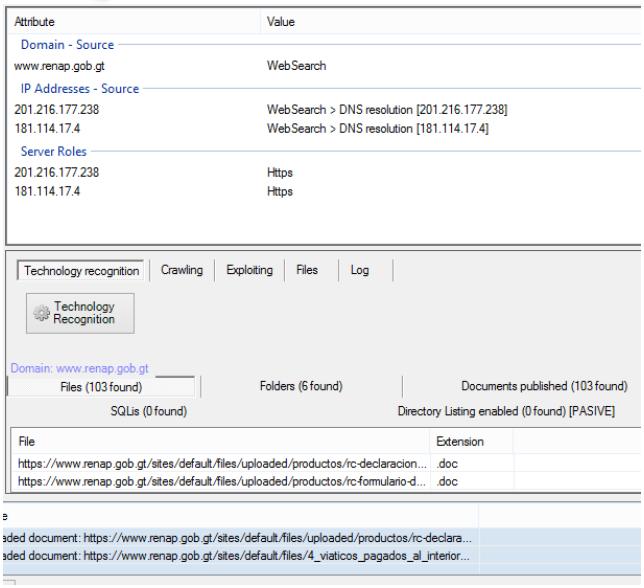
Hallazgo



Hallazgo	Recomendación
Al realizar análisis de Metadatos del sitio web de RENAP: **https://www.renap.gob.gt/** determinamos que es posible obtener archivos sensibles como el archivo siguiente archivo. FILETYPE:DOC Nombre: Declaración jurada testigo no puede firmar	El Administrador web debe hacer uso de estándares de programación de calidad. El cual nos ayuda a la declaración correcta de variables, clases, métodos, funciones. Dentro de los estándares, debemos obtener la capacidad máxima para el uso de **.htaccess**, donde podemos controlar determinados aspectos de nuestro sitio web, incluyendo el rendimiento.
	También con la técnica de iframe. Se trata de utilizar la etiqueta <iframe> con algunas modificaciones, y con el agregado de algunas propiedades CSS. Ej. <iframe src="xxxxxxxxx.html"></iframe>

Mapa de Servidores

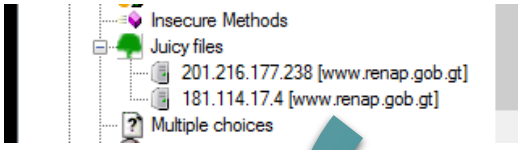


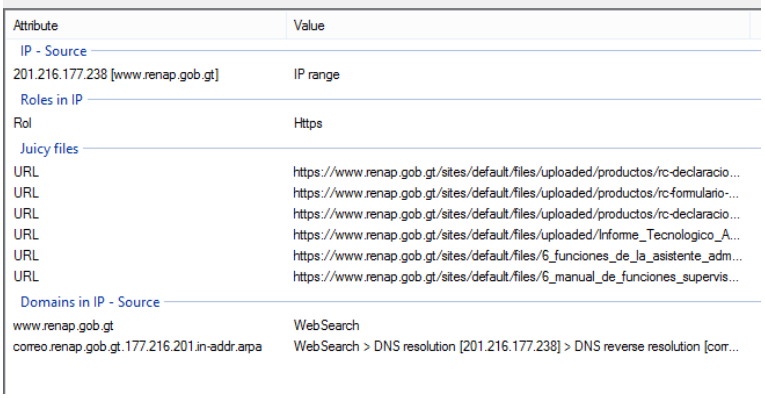


Hallazgo	Recomendación
Al realizar análisis de Metadatos del sitio web de RENAP: **https://www.renap.gob.gt/** determinamos que el acceso al mapa de los servidores del sitio web.	Se debe realizar las respectivas correcciones en cuanto a configuraciones del acceso del NOS de los servidores. Ej. !========== !Configuraciones Basicas para R1 !========== enable configure terminal hostname Router_Renap_1 no ip domain-lookup ! enable secret AS_security #banner motd #Prohibido el acceso sin autorización, sistema sujeto a monitoreo solo personal autorizado# ! line con 0 password cisco login loggin synchronous ! line vty 0 4 password AS_security_ login loggin synchronous ! end write

Vulnerabilidades Encontradas:

- **Juicy Files:** Todos esos ficheros suelen contener datos jugosos, y en la FOCA existen listas para clasificarlos cuando sean encontrados, son los "Juicy Files". Es configurable cuándo debe ser tomado un fichero de este tipo, y dependerá de cada proyecto, pero por defecto lo será cualquier fichero con una extensión que no sea de las comunes.

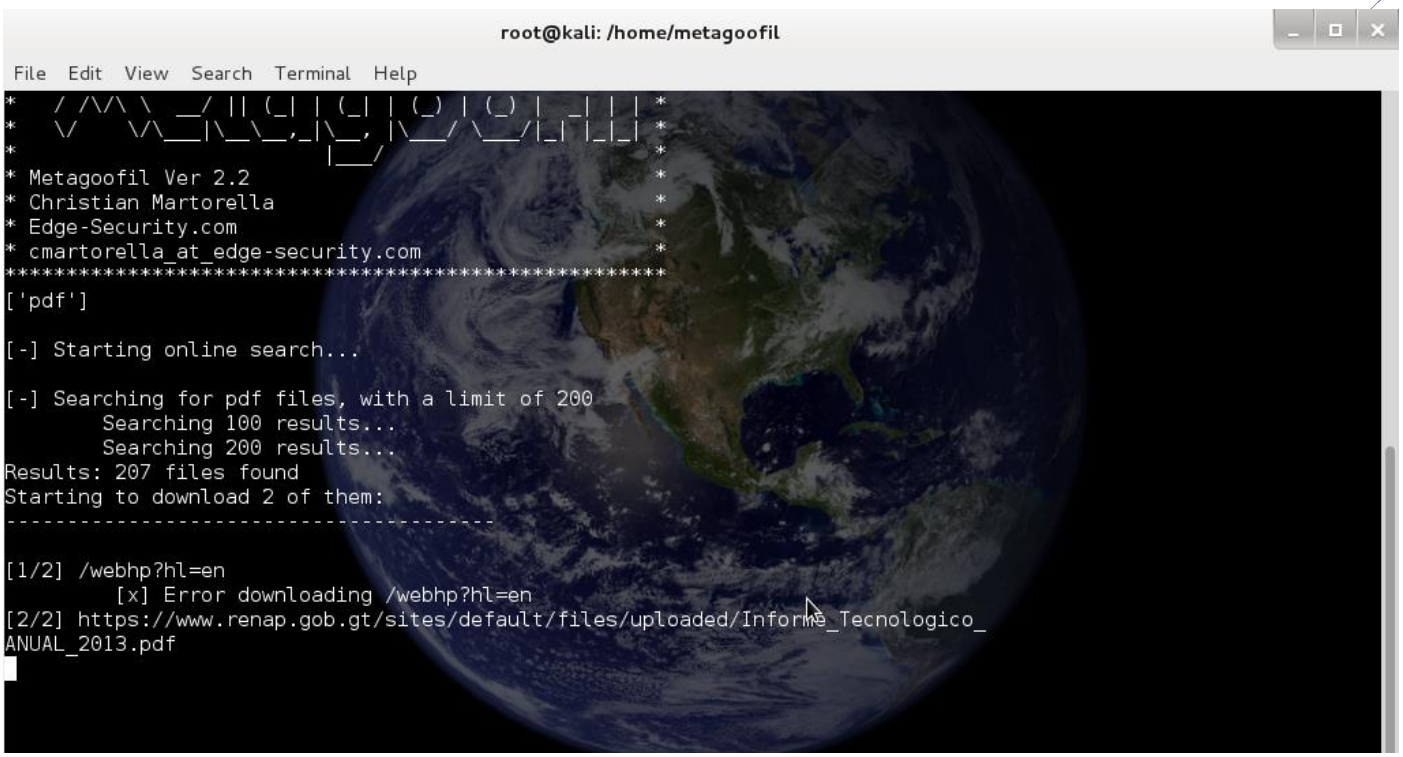




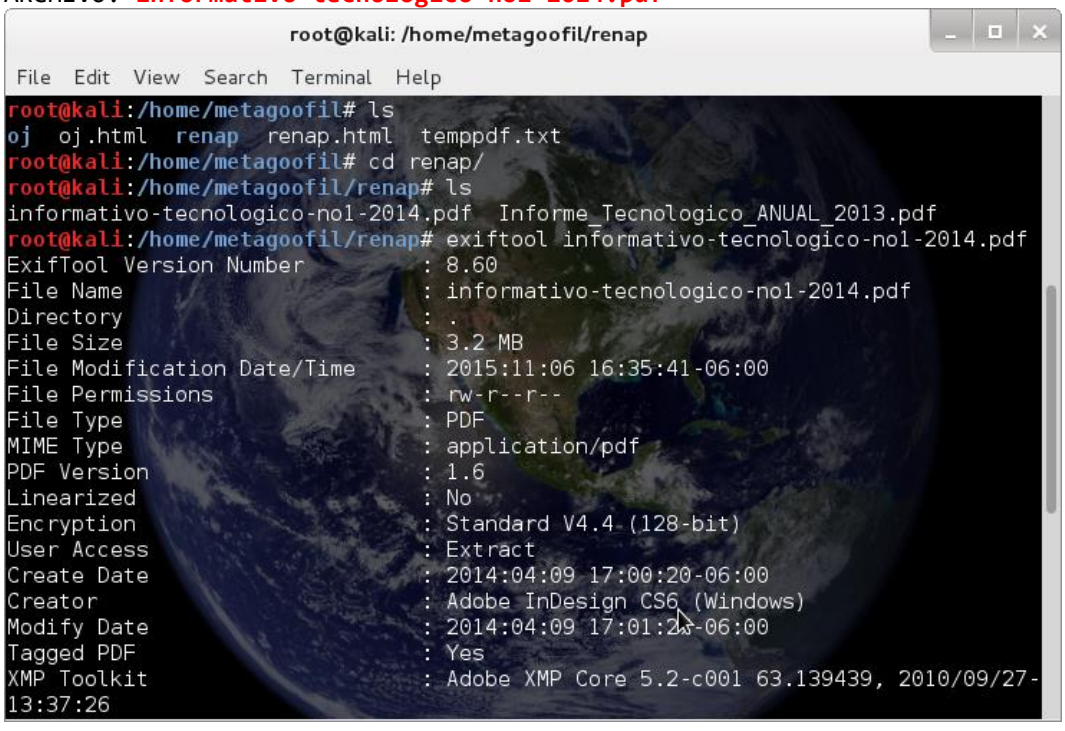
Hallazgo	Recomendación
Al realizar análisis de Metadatos del sitio web de RENAP: **https://www.renap.gob.gt/** determinamos vulnerabilidades de tipo Juicy Files.	Hacer uso de estándares a la hora de nombrar los archivos, sabiendo que éstos pueden ser usados por terceros. Ej. Renap_AS1.pdf

Seguridad de Datos sitio Web: https://www.renap.gob.gt/

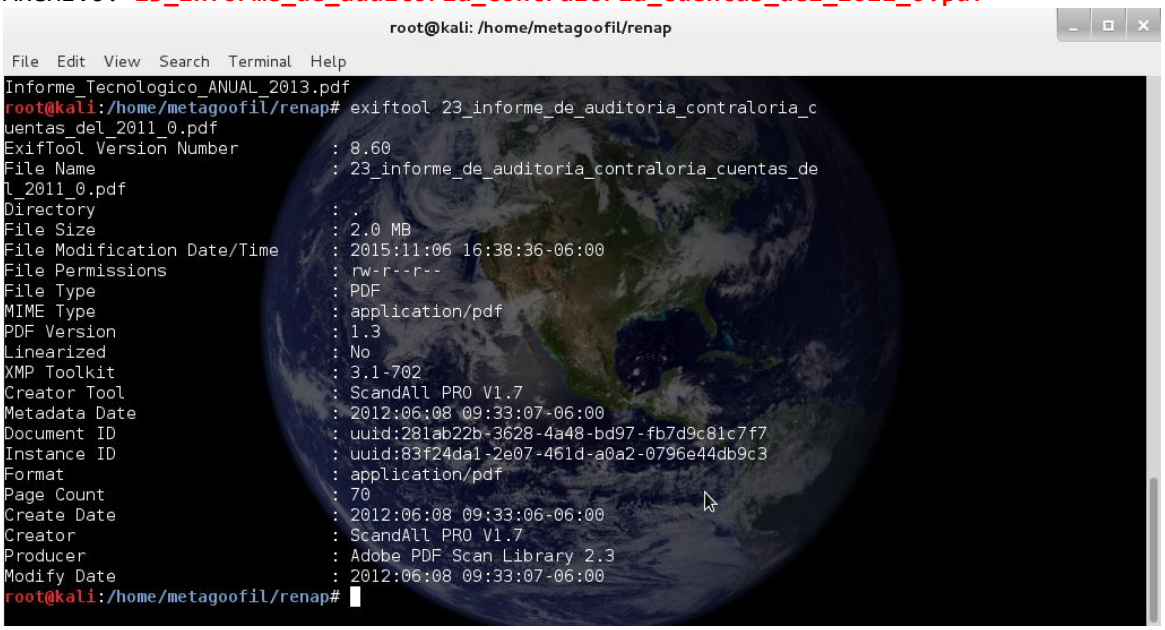
Herramienta Utilizada: METAGOOFIL (KALI)



Seguridad de Sitio Web : https://www.renap.gob.gt/
Herramienta Utilizada: EXIFTOOL (KALI)
ARCHIVO: **informativo-tecnologico-no1-2014.pdf**



ARCHIVO: **23_informe_de_auditoria_contraloria_cuentas_del_2011_0.pdf**

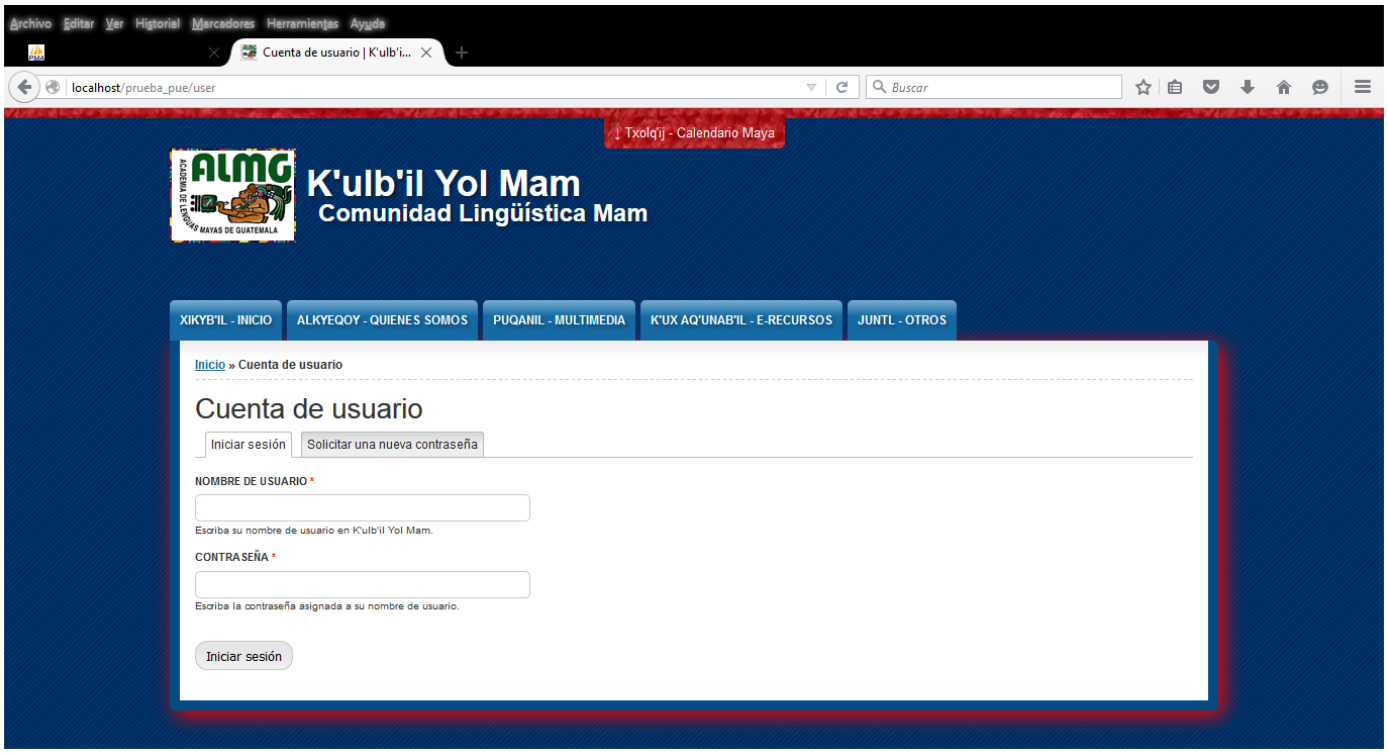


Hallazgo	Recomendación
Al obtener metadatos de archivos, puede ser sumamente dañino para el Sistema de Información de nuestra empresa, por lo que se debe de evitar que nuestros archivos tengan enriquecido los metadatos.	Tenemos las siguientes aplicaciones de forma gratuita para la eliminación de metadatos. • **MetaStripper 0.92** • **MetaClean 2.7.2** • **BatchPurifier 3.1**

Seguridad de contraseñas en un sitio web:

CMS: DRUPAL 7.3

Encriptación: Simple md5 hex. Hash

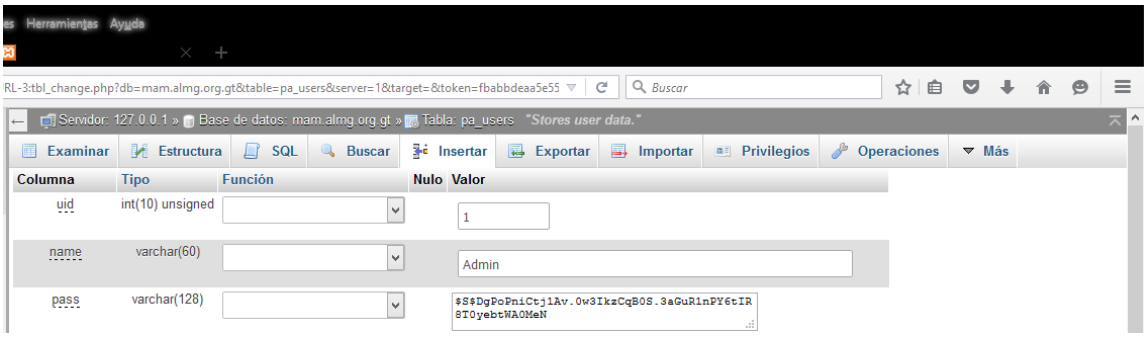


A continuación modificaremos la contraseña de administrador:

Primero: Modificamos el archivo index.php con el siguiente código:

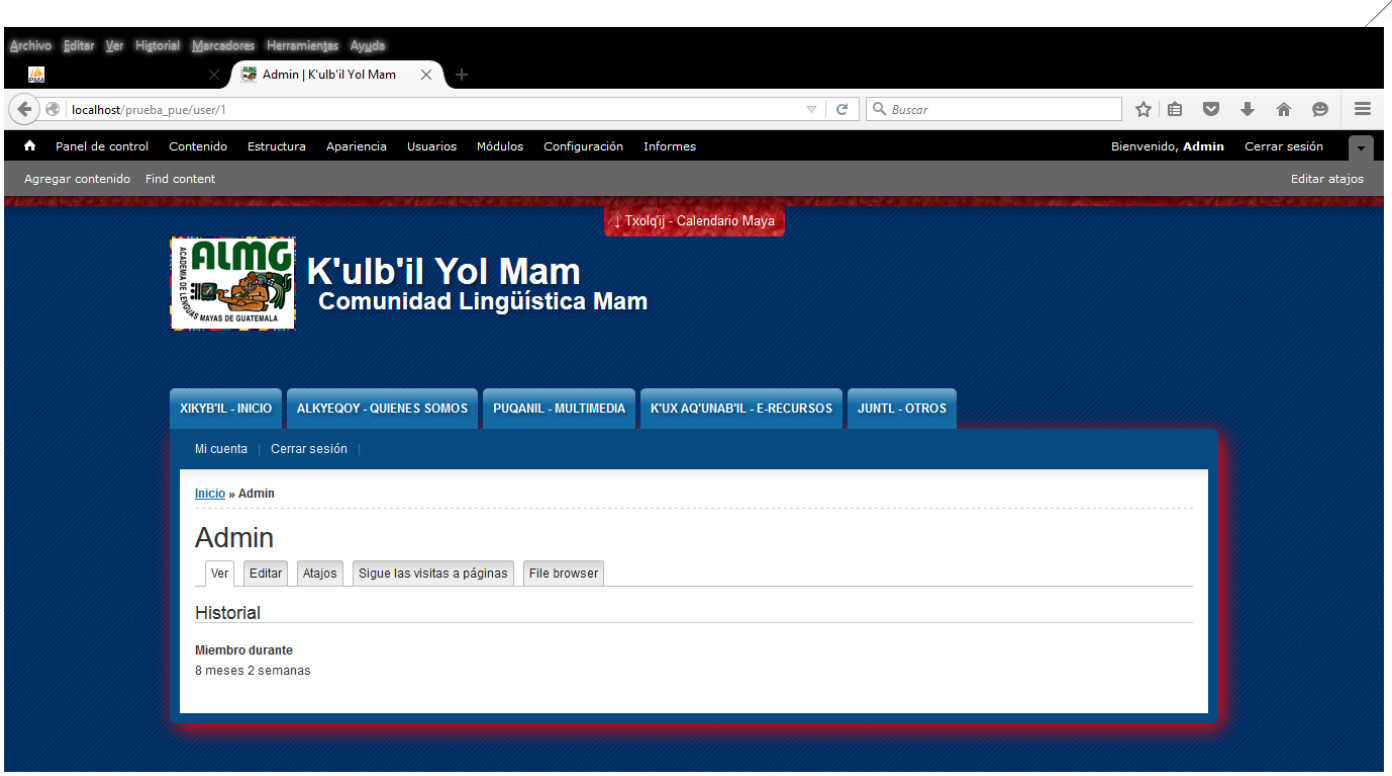
```
define('DRUPAL_ROOT', getcwd());

require_once DRUPAL_ROOT . '/includes/bootstrap.inc';
drupal_bootstrap(DRUPAL_BOOTSTRAP_FULL);
menu_execute_active_handler();

DESPUÉS:

define('DRUPAL_ROOT', getcwd());

require_once DRUPAL_ROOT . '/includes/bootstrap.inc';
drupal_bootstrap(DRUPAL_BOOTSTRAP_FULL);
require('includes/password.inc');
echo user_hash_password('PUE_2015');
die();
menu_execute_active_handler();
```

Modificamos la tabla user de la base de datos:



Rectificamos el archivo index.php a su estado normal.

Y luego intentamos logearnos como administradores con la contraseña que indicamos anteriormente: PUE_2015



Es importante velar por el cumplimiento de normas que impidan este tipo de errores de programación, para ofrecer una mejor seguridad de nuestros datos.