ARTÍCULO DE REVISIÓN

Consideraciones para mejorar la seguridad en los sistemas gestores de contenido (cms) Joomla!

Considerations to improve the security in Joomla! System manager content (cms)

M.Sc. Gerardo Junco Romero,[I] Dra. Sonia Rabelo Padua[II]

Instituto Nacional de Salud de los Trabajadores (INSAT). Cuba. E-mail: gerardo@insat.sld.cu
Instituto Nacional de Salud de los Trabajadores (INSAT). Cuba. E-mail: sonia@insat.sld.cu

RESUMEN

El desarrollo extraordinario de la tecnología, los resultados de la innovación tecnológica y más aún en el campo de las ciencias informáticas y las telecomunicaciones, hacen de la realización de proyectos de investigación algo cotidiano en las organizaciones empresariales e instituciones públicas. La mayoría de las veces los resultados de estas investigaciones, así como otras informaciones de carácter general son presentados en plataformas web, haciendo uso de gestores de contenido por las ventajas que ofrecen estos con respecto a su alcance y fácil manipulación para mantenerlos actualizados, evidenciándose en una mayor productividad de la entidad al emplearlos. Es por esta razón que se hace imprescindible que las instituciones se interesen por las redes de computación, el mantenimiento y la seguridad requerida para la utilización de los gestores de contenido. Con el fin de prestar atención a este último aspecto (la seguridad), es que el presente trabajo aborda una serie de aspectos que tanto programadores, como administradores de red deben manejar para mejorar la toma de decisiones a la hora de proteger la información expuesta en Web basadas en sistemas gestores de contenido, y en específico aquellas basadas en Joomla!

Palabras claves: seguridad informática, sistema gestor de contenidos, Joomla!

ABSTRACT

The extraordinary development of technology, the results of technological innovation and even more in the field of computer science and telecommunications, make the realization of research projects something every day in business organizations and public institutions. Most of the time the results of this research, as well as other general information are presented in web platforms, making use of content managers because of the advantages they offer with respect to their scope and easy manipulation to keep them updated, evidenced in greater productivity of the entity when using them. It is for this reason that it is essential that institutions are interested in the computer networks, maintenance and security required for the use of content managers. In order to pay attention to this last aspect (security), it is that the present work shows a series of aspects that both programmers and network administrators must manage to improve decision making in protecting the information exposed in Web-based content management systems, and specifically those based on Joomla!

Key words: information security, content management system, Joomla!

INTRODUCCIÓN

Conformar una red telemática integrada y apoyada en las más modernas tecnologías, tanto en hardware como en software, ha sido siempre la misión del colectivo de informática, para así contribuir al mejoramiento de la investigación, la docencia y la gestión de la información de forma general en el Instituto Nacional de Salud de los Trabajadores (INSAT).

Los sitios Web son una de las variantes más empleadas a nivel mundial para difundir información, donde a nivel empresarial el objetivo es integrar los servicios que se brindan, así como mostrar los resultados alcanzados por la institución, entre otros ejemplos.[1] Este servicio emplea un programa denominado servidor Web, el cual se basa en la tecnología cliente – servidor, encargándose de gestionar las peticiones en el lado del servidor realizando conexiones bidireccionales y/o unidireccionales con el cliente, generando una respuesta en el lado del cliente.[2] El código recibido por el cliente suele ser interpretado y ejecutado por un navegador Web. Para la transmisión de todos estos datos se utiliza generalmente el protocolo HTTP.

En la actualidad, el uso de sistemas gestores de contenidos (CMS) agiliza el proceso de implementación de Webs institucionales en comparación con métodos más tradicionales, los cuales implican más tiempo de desarrollo y conocimientos específicos de programación, pues existen disímiles productos que se ajustan en mayor o menor medida a las necesidades puntuales de cada institución, o sea, algunos gestores de contenidos están enfocados a la gestión de información mediante blogs, al aprendizaje autodidacta, al marketing, entre otros.[3] Además otra de las ventajas que ofrecen estos gestores de contenidos es que permiten configurarse de forma tal que el programador o webmaster encargado de la Web pueda personalizarlo a su gusto y necesidades objetivas de la institución, todo a partir de una interfaz de administración muy intuitiva, no demandando

conocimientos específicos o avanzados en materia de programación Web, lo cual no minimiza el hecho de que aquellas personas con un dominio amplio en esta materia puedan lograr personalizaciones más rápidas y avanzadas sobre aquellos usuarios que no la dominen.

La seguridad de toda la información que se procese, intercambie y se reproduzca a través de estos CMS es siempre objetivo de debates en la red de redes,[4] pues algunos plantean que es difícil garantizarla mediante la administración que estos sistemas ofrecen, siendo a veces esta última (la administración del CMS) su punto más vulnerable. Independientemente de la postura adoptada, no se puede negar que detrás de casi todos los sistemas gestores de contenidos existe una gran comunidad de programadores que los emplea, y por tanto también aportan soluciones novedosas tanto ampliando las funcionalidades y servicios que pueden brindar, así como mejoras en la seguridad del propio sistema. El presente trabajo aborda una serie de aspectos que debemos considerar si se quiere optimizar la seguridad de los sistemas gestores de contenidos, en específico aquellos que estén basados en Joomla! OBJETIVO: Proponer un conjunto de acciones que contribuyan al mejoramiento de la seguridad en los sistemas gestores de contenidos basados en Joomla! MATERIAL Y MÉTODOS Se realizó un análisis documental a partir de información recuperada en materiales impresos (artículos en revistas especializadas, tesis de grado, entre otros) así como en Internet bajo los descriptores: sistemas gestores de contenidos, seguridad, Joomla!. A partir de la información analizada y la experiencia vivencial de los autores se elaboró una propuesta de consideraciones a tener en cuenta para optimizar de forma general la seguridad en los gestores de contenidos y en específico a aquellos que estén basados en el CMS Joomla! RESULTADOS Elementos y medidas de seguridad en Joomla!

Para aumentar el nivel de seguridad en el CMS Joomla! se pueden realizar algunos ajustes. Hay que partir de la base que Joomla es seguro, siempre que se realice un buen mantenimiento a nivel de actualizaciones y se tomen ciertas medidas de precaución, tanto a nivel de servidor como del propio Joomla!.[5]

Fundamentalmente hay 3 aspectos que deben contemplarse cuando hablamos de seguridad Joomla:

1. La configuración del servidor donde está alojado Joomla!
2. Las medidas tomadas desde la administración del Joomla por parte del administrador, tanto a nivel de configuraciones como extensiones de seguridad.
3. Extensiones de terceros, hay que tener en cuenta su seguridad y mantenerlas actualizadas.

Consideraciones generales

- Cambie sus contraseñas regularmente y no use siempre las mismas para todos los servicios que usted administra. Utilice una combinación aleatoria de letras, números, o símbolos y evite usar nombres o palabras que puedan ser encontradas en un diccionario. Nunca utilice los nombres de sus parientes, mascotas, etc.

- Si usted esta usando un servicio compartido de hospedaje en su proveedor, asegúrese de que ningún otro usuario en el servidor pueda ver o acceder a los archivos de su sitio, por ejemplo a través de cuentas shell, cpanels, etc.

- Nunca dependa de los backups de otro. Hágase responsable personalmente de respaldar regularmente los archivos de su sitio y su base de datos.

- Utilice un sistema de Prevención/Detección de intrusos (IDS) para bloquear/alertar sobre solicitudes HTTP maliciosas.

Consideraciones para el servidor donde se hospeda el CMS:

Es recomendable siempre trabajar cualquier actualización, implementación de nuevas funciones, así como mejoras de seguridad en un servidor de desarrollo local, de forma tal que pueda probar en este el correcto funcionamiento de lo que implemente, garantizando así que la Web no pierda visibilidad por mal funcionamiento de nuevas mejoras. Para ello se puede emplear un instalador de aplicaciones LAMPP fácil de usar y gratuito que trabaja en muchos sistemas operativos, incluyendo GNU/Linux y Windows.[6,7]

Al igual que la mayoría de las Webs, los gestores de contenidos (CMS) están basados en una estructura de páginas dinámicas y una base de datos que se encarga de guardar las configuraciones del sitio, así como de los contenidos del mismo.[8,9] Es por ello que a la hora de implementar acciones para mejorar la seguridad de los CMS debemos tomar medidas en ambos servicios de red (servidor web y servidor de base de datos). Por lo general en redes pequeñas estos servicios cohabitan en el mismo servidor, pero independientemente de ello, las medidas que a continuación se ofrecen son válidas para cualquiera de las dos variantes.

Servidor Web (HTTP Server):[10]

- PHP, MySQL y muchos otros componentes base fueron originalmente diseñados para, y generalmente funcionan mejor en, servidores Apache. Evite usar otros servidores si es posible.

- Utilice archivos .htaccess para bloquear intentos de exploits.

- Regularmente revise los registros de acceso en busca de actividad sospechosa. No confié en sumarios y graficas. Revise los "raw logs" (registros en crudo) para detalles más reales.

- Configure los filtros de Apache mod_security y mod_rewrite para que bloqueen ataques PHP.

Servidor de Base de Datos (MySQL):

- Asegúrese de que la cuenta MySQL de Joomla! está configurada con acceso limitado. Esté consciente de que la instalación inicial de MySQL es insegura. Una cuidadosa configuración manual es requerida luego de la instalación.

- En un servidor compartido, si usted puede ver los nombres de las bases de datos de otros usuarios, entonces puede estar bastante seguro de que ellos ven las suyas. Si ellos pueden ver las bases de datos que usted posea, ellos están innecesariamente un paso más cerca de entrar. Un buen proveedor de servicios limitará estrictamente el acceso de cada usuario a sus propias bases de datos.[11,12]

PHP:

El CMS Joomla! al igual que muchos otros, está programado en PHP, por lo cual en aras de optimizar la seguridad del CMS a continuación ofrecemos una serie de recomendaciones para ello:[13,14,15]

- Aplique todos los parches necesarios para PHP y para aplicaciones basadas en PHP.

- Se recomienda un frecuente escaneo dónde un gran número de aplicaciones PHP están en uso.

- Utilice herramientas como Paros Proxy para realizar pruebas automáticas de SQL Injection en contra de sus aplicaciones PHP.

- Siga el principio de "Least Privilege" (El menor privilegio) para hacer funcionar PHP usando herramientas como PHPsuExec o suPHP.

El fichero de configuración php.ini

- Estudie la lista oficial de directivas php.ini en http://www.php.net

- Configure register_globals OFF. Esta directiva determina si registrar o no las variables EGPCS (Environment, GET, POST, Cookie, Server) como variables globales.

- Use disable_functions para desactivar peligrosas funciones PHP que no son necesarias para su sitio.

- Desactive allow_url_fopen. Esta opción activa las URL-aware fopen wrappers que permite el acceso a los objetos URL como archivos. Los wrappers (envolturas) son proveídos para el acceso de archivos remotos usando el ftp o el protocolo http, algunas extensiones como zlib son capaces de registrar wrappers adicionales. Esto solo puede ser configurado en php.ini por motivos de seguridad.

- Ajuste la directiva magic_gpc_quotes como sea necesario para su sitio. Debería estar en off para usar software bien escrito, y para los pobremente escritos scripts PHP 3 y PHP 4 . magic_gpc_quotes configura el estado magic_quotes state para operaciones GPC (Get/Post/Cookie). Cuando magic_quotes esta on, todas las ' (single-quote/comillas-simples), " (double quote/comillas dobles), \ (backslash-barra invertida) y NUL's son evitadas con una barra invertida \ automáticamente.

-Modo Seguro: safe_mode (debería estar activado y configurado correctamente)

- open_basedir (debería estar activado y configurado correctamente). Limite los archivos que pueden ser abiertos por PHP al árbol de directorios especificado, incluyendo el archivo mismo. Esta directiva no es afectada si el Safe Mode esta On u Off. La restricción especificada con open_basedir es en realidad un prefijo, no un nombre de directorio. Esto significa que "open_basedir = /dir/incl" también permite el acceso "/dir/include" y "/dir/incls" si es que existen. Cuando quiere restringir el acceso solamente al directorio especificado, cierre con una barra /.

Directivas de ejemplo para las sugerencias anteriores:[16]
register_globals = 0
disable_functions = show_source, system, shell_exec, passthru, exec, phpinfo,

popen, proc_open
allow_url_fopen = 0
magic_gpc_quotes = 0
safe_mode = 1
open_basedir = /dir/incl/

Consideraciones para el núcleo de Joomla! (CORE)

- Siempre actualice a la última versión estable.

- Descargue Joomla! solo de sitios oficiales, sitios de confianza.

- Suscríbase, o revise regularmente anuncios relacionados con la seguridad Joomla!

- Remueva todas las plantillas (templates) que no sean necesarias en su sitio.

- Edite globals.php y establezca register_globals emulation en off para Joomla!. Aunque la emulación Joomla! es mucho más segura que la directiva PHP register_globals, es mejor no permitir para nada register_globals.

- Una vez que su sitio está configurado y es estable, proteja contra escritura la mayor cantidad de archivos y directorios que pueda cambiando los permisos de directorios a 755, y los permisos de archivos a 644.

- Existe una característica de sitio en Global Configuration (configuración global) que puede colocar los permisos de forma masiva por usted. Tenga en cuenta de que esta función masiva puede afectar el funcionamiento de los componentes, si lo hace pruebe el funcionamiento de los mismos. También tenga en cuenta que es posible que no se puedan cambiar los permisos en todos los componentes o extensiones.

Consideraciones para las extensiones **(Componentes, Módulos, y plugins) de Joomla!**

- No utilice extensiones Joomla! que requieran register_globals ON.

- Descargue extensiones solo de sitios de confianza. La definición oficial de "sitio de confianza" es aquel sitio en el que USTED confía.

- Independientemente de que se hagan copias de seguridad planificadas con la frecuencia estipulada por el administrador del sitio, se recomienda que siempre se realice una copia de seguridad de su sitio y de la base de datos MySql, antes de instalar nuevas extensiones, de esta forma se garantiza que en caso de que no funcione correctamente el componente instalado, se utilice la misma como un punto de restauración anterior, en el cual la pérdida de información sea mínima o nula.

- Desinstale cualquier extensión no usada, y revise doblemente que los directorios y archivos relacionados hayan sido borrados.

Respecto a la instalación de componentes que eleven el nivel de seguridad de Joomla, es recomendable utilizar:

SecureJoomla:

jSecure Authentication. Para poner una barrera de seguridad adicional a nuestro sitio, este plugin modifica la ruta de acceso a la administración. Se trata de añadir una clave personal a nuestra ruta del administrador, ya que debido a que la ruta de acceso a la administración de Joomla! es siempre la misma: (http://www.misitio.com/administrator), cualquiera que haya logrado obtener nuestro usuario o contraseña puede acceder a nuestro Joomla!

CONCLUSIONES

El empleo de sistemas gestores de contenidos es hoy una necesidad para la mayoría de las instituciones empresariales, por las ventajas que ofrece en cuanto a rapidez para obtener un producto terminado y con la calidad requerida de acuerdo con los estándares que demanda el mundo actual, así como otras ventajas referidas a la estabilidad, escalabilidad y fácil manipulación del sistema gestor en general.

La sistematización realizada de los sistemas gestores de contenidos (CMS) permitió analizar los elementos básicos que deben tenerse en cuenta para optimizar la seguridad de la información que se procesa, intercambia y se reproduce a través de estos sistemas.

El sitio web de la intranet del INSAT está elaborado con el sistema gestor de contenidos Joomla! y en este han sido probadas e implementadas las medidas descritas en este artículo, permitiendo así alcanzar un mayor desarrollo de las TIC y de la cultura informática del personal de la institución, enmarcándose en un contexto tecnológico y sociocultural, donde se hace necesario, una correcta selección de las herramientas y dispositivos a emplear dentro de la red informática, en función de optimizar los recursos y la propia infraestructura para la cual fue diseñada.

REFERENCIAS BIBLIOGRÁFICAS

1. Área M. "De los Web educativos al material didáctico Web". Artículo publicado en la revista COMUNICACIÓN Y PEDAGOGÍA. Universidad de La Laguna. España. 2003.

2. Fiton JM. Arquitectura de Redes. UTE de Bayonne, Noviembre 2000.

3. Atlet I. "Las tecnologías de la informática y las comunicaciones como factores de éxito en la nueva economía. América Latina en el contexto global". Ponencia presentada en el XXIII Congreso Internacional de la Asociación de Estudios Latinoamericanos, LASA, 2001, Washington DC, Sep. del 6 al 8 2001. p12.

4. Castillo E. "La comunicación y la cibernética", en Revista Latina de Comunicación Social, número 17, de mayo de 1999, La Laguna (Tenerife). Consultado el 23 de enero de 2015. Disponible en: http://www.ull.es/publicaciones/latina/a1999hmy/90emilce.htm

5. Hasan Y, Hernandez J. "Diseño de Arquitecturas de Información: Descripción y Clasificación". Consultado el 23 de septiembre de 2016. Disponible en: http://www.nosolousabilidad.com/articulos/descripcionyclasificación

6. El sistema operativo de linux como servidor. Consultado el 23 de enero de 2015. Disponible en: http://elmejorhostingbarato.com/sistema-operativo-linux/

7. DIEC. Server Oriented Operating System. Consultado el 25 de mayo de 2010. Disponible en: http://www.ingelec.uns.edu.ar/rts/soos/

8. Bueno M. "Generación dinámica de sitios Web". UHC; Habana Dic 2002.

9. Blanco LJ. "Auditoria a sitios Web". GIGA Revista editada por COPEXTEL Cuba, Numero 2/ 2003.

10. PHP. Historia de PHP y Proyectos Relacionados. Consultado el 17 de octubre de 2014. Disponible en: http://web.archive.org/web/http:/es2.php.net/history

11. Tanenbaum AS. Redes de ordenadores (Google Books) (4ª edición). Pearson Educación. ISBN 9789702601623. Consultado el 26 de enero de 2015. Disponible en: http://books.google.es/books?id=WWD-4oF9hjEC

12. Dalmau M, Redes de Computadoras. UTE de Bayonne, Noviembre 2000.

13. PHP 7.0.13 Released. PHP (2016). Consultado el 10 de noviembre de 2016. Disponible en: https://php.net/archive/2016.php#id2016-11-10-1

14. PHP 7.1.0 Release Candidate 6 Released». PHP (2016). Consultado el 10 de noviembre de 2016. Disponible en: https://php.net/archive/2016.php#id2016-11-10-2

15. W3C. Extensible Markup Language (XML). W3C. (2003) http://www.w3.org/TR/REC-xml

16. PHP. PHP: Sintaxis básica. Consultado el 13 de abril de 2015. Disponible en: http://php.net/manual/es/language.basic-syntax.php

Recibido: 25 de noviembre de 2016.
Aprobado: 13 de marzo de 2017.