Revista Cubana de Ciencias Informáticas
Vol. 12, No. 1, Enero-Marzo, 2018
ISSN: 2227-1899 | RNPS: 2301
Pág. 43-57
http://rcci.uci.cu

Tipo de artículo: Artículo original
Temática: Seguridad Informática
Recibido: 11/12/2017 | Aceptado: 22/01/2018

Estudio de patrones de intentos de ciberataques asociados a las vulnerabilidades del complemento RevSlider

Study of cyber attack attempts patterns associated with vulnerabilities of the RevSlider plugin

Henry Raúl González Brito[*]

Dirección de Seguridad Informática, Universidad de las Ciencias Informáticas. Carretera a San Antonio de los Baños, Km. 2 ½. Torrens, municipio de La Lisa. La Habana, Cuba.

[*] **Autor para correspondencia**: henryraul@uci.cu

Resumen

En el trabajo se describen los patrones de intentos de ciberataques contra aplicaciones web basadas en WordPress a través del componente RevSlider. La presencia de versiones desactualizadas y vulnerables de RevSlider ha comprometido cientos de miles de sitios desde el año 2014. El estudio se realizó a través del análisis de trazas de acceso desde Internet de dominios y aplicaciones web a lo largo de diez meses. Se describe el nivel de intensidad de los intentos de ciberataques por meses, las rutas bases utilizadas, los nombres de recursos empleados, los temas de WordPress explorados y el empleo de direcciones IP. Se concluye que ninguna aplicación web estudiada fue comprometida y que se hace necesario mantener actualizaciones regulares de la base tecnológica de las aplicaciones web.

Palabras clave: aplicaciones web, ciberataques, complementos, RevSlider, WordPress

Abstract

The article describes the patterns of attempts of cyber attacks against web applications based on WordPress through the RevSlider plugin. The presence of outdated and vulnerable versions of RevSlider has compromised hundreds of thousands of sites since the year 2014. The study was conducted through the analysis of Internet access traces of domains and web applications over ten months. It describes the intensity level of the attacks for months, the base routes used, the names of resources used, the WordPress themes scanned and the use of IP addresses. It is concluded

that no studied web application was compromised and that it is necessary to maintain regular updates of the technological base of web applications.

Keywords: *cyber attacks, plugins, RevSlider, web applications, WordPress*

Introducción

Las aplicaciones web representan una gran parte de los servicios informáticos disponibles en la sociedad moderna. Por este motivo se incrementa la necesidad de garantizar niveles de seguridad apropiados en ellas, las cuales, por causa de su crecimiento, son el blanco preferido de delincuentes, perjudicando a usuarios y organizaciones (KENT *et al.*, 2016; VASEK *et al.*, 2016).

Las vulnerabilidades son defectos o debilidades en el diseño, implementación, funcionamiento o administración de los sistemas informáticos que pueden ser usados para comprometer los requisitos de seguridad. Son la principal causa de que las aplicaciones web puedan ser atacadas con éxito por los ciberdelincuentes (VIANO, 2017). Reportes de organizaciones y gobiernos alrededor de todo el mundo coinciden en este hecho (OWASP, 2013; SINGHAL y OU, 2017).

Las vulnerabilidades pueden encontrarse en cualquier parte de las aplicaciones web, incluyendo los complementos, también conocidos por plugins, que se utilizan para proveerle funcionalidades adicionales (JERKOVIĆ *et al.*, 2016; KAUR *et al.*, 2017). En los reportes de la empresa consultora Sucuri del 2016 (SUCURI, 2016), se refleja que el 22 % de las aplicaciones web afectadas, basadas en el CMS (Content Management Systems) WordPress, tenían complementos con vulnerabilidades conocidas (da Fonseca and Vieira, 2014; Vahdani Amoli, 2015; Zhou, Gu et al., 2015; Riadi and Aristianto, 2016; Vasek, Wadleigh et al., 2016). Solamente uno de estos complementos, nombrado RevSlider, permitió que fueran comprometidas el 10% de las aplicaciones web basadas en WordPress.

Los reportes de reconocidas empresas a nivel internacional e instituciones gubernamentales dedicados a ciberseguridad, son la principal fuente para analizar el comportamiento de los ciberataques a nivel mundial (HUANG *et al.*, 2016). En sentido general, estos reportes carecen de vistas que permitan comprobar el comportamiento de los ciberataques como un proceso continuo en el tiempo (CARVER *et al.*, 2016). Estas deficiencias impiden que los desarrolladores y administradores de aplicaciones web comprendan que las aplicaciones web, sobre todo aquellas

publicadas en Internet, son blanco continuo de ciberataques automatizados que exploran permanentemente las vulnerabilidades recientes con el objetivo de explotarlas (LALLIE *et al.*, 2017).

El objetivo principal de este trabajo es describir el comportamiento de los intentos de explotación de las vulnerabilidades del componente RevSlider de WordPress. Mediante este estudio, los especialistas de ciberseguridad y desarrolladores de software, podrán apreciar los mecanismos de reconocimiento y explotación sistemática que se ponen de manifiesto en Internet contra los dominios asociados a aplicaciones web para aplicar las medidas correspondientes que fortalezcan la seguridad de las aplicaciones web basadas en WordPress.

Materiales y métodos o Metodología computacional

Para la realización de la investigación se analizaron los registros de accesos desde Internet, a diferentes sitios y aplicaciones web entre los meses de marzo y diciembre del 2016. Estos registros almacenan un conjunto de datos de acceso como la fecha y hora, URL, código de respuesta HTTP y otros campos de encabezados de petición HTTP.

Se realizó un proceso de limpieza y transformación de los registros con el objetivo de extraer los datos útiles para la investigación. Para ello se escribieron varios conjuntos de instrucciones con el intérprete de comandos Bash y se utilizaron programas como cat, grep, awk, cut, entre otros. Los datos de interés se escribieron en el formato CSV para un análisis posterior en hojas de cálculo.

La URL solicitada se dividió en tres partes para su estudio del siguiente modo: URL = Dominio + Ruta + Recurso donde:

- Dominio: Es la parte inicial de la cadena hasta la primera barra invertida (/) si estuviera presente. Es donde se establece el dominio bajo el cual está desplegado una aplicación web o un servicio de cualquier tipo. Por ejemplo www.example.com.
- Ruta: Es la cadena que indica la dirección donde se encuentra el recurso solicitado y aparece a la derecha de la barra invertida (/). Por ejemplo /wp-content/plugins/RevSlider/. Se incluye bajo este concepto también los recursos a los cuales se les envían los parámetros de la URL.
- Recurso: Es el elemento solicitado bajo el dominio y ruta especificada. Por ejemplo index.php y readme.txt.

Resultados y discusión

Vulnerabilidades asociadas al componente RevSlider

El componente RevSlider, también conocido por Slider Revolution, brinda funcionalidades de transición de imágenes en las aplicaciones web con el objetivo de hacerlas más agradable al usuario y adaptar la interfaz visual a diferentes tamaños de pantalla. RevSlider puede utilizarse directamente o como parte de un tema del sistema de gestión de contenidos WordPress. Fue Liberado el 31 de julio del 2012 por la compañía ThemePunch (www.themepunch.com). Es un complemento de pago, con más de 215 267 ventas solamente en la codecanyon.net. Más de 4 millones de aplicación web lo siguen utilizando actualmente.

En el 2014 fue descubierta una vulnerabilidad en este componente de tipo Local File Inclusion (LFI) o Inclusión de Archivos Locale (MUSCAT, 2016; STAROV *et al.*, 2016). Las vulnerabilidades LFI permiten el acceso, carga y ejecución de archivos en la aplicación web, debido a la dependencia de parámetros de entradas con referencias a archivos que pueden ser manipulados por los ciberatacantes (LE *et al.*, 2016). Un código vulnerable (CHIESA y DE LUCA SAGGESE, 2016) típico escrito en el lenguaje de programación PHP es semejante a este:

```
$file = $_GET['file'];
if(isset($file))
{
   include("opcion/$file");
}
else
{
   include("index.php");
}
```

Como puede notarse, este código recibe como entrada el parámetro file, conteniendo la referencia a un archivo que va a ser incluido en el flujo de ejecución del programa. Un ciberatacante por tanto, puede manipular el parámetro file para forzar la inclusión de un archivo en el proceso de ejecución en el servidor web, ya sea un archivo del sistema como /etc/passwd para la exposición de su contenido o la ejecución de un código dañino en forma de puerta trasera o shellcode (OKAMOTO y TARAO, 2016; WANG *et al.*, 2017). También es posible la inyección de un shellcode a través del cuerpo de una petición POST (MEDEIROS *et al.*, 2016; RIADI y ARISTIANTO, 2016). Un ejemplo expuesto es la siguiente petición HTTP:

http://ejemplo.com/index.php?file=../../../etc/passwd

Las vulnerabilidades asociadas a RevSlider han sido recogidas bajos los identificadores CVE-2014-9734, CVE-2014-9735, CVE-2015-5151 y CVE-2015-1579. Akamai reportó que en el primer trimestre del año 2015, se utilizaron estas vulnerabilidades para afectar el 66% de las aplicaciones web comprometidas (AKAMAI_TECHNOLOGIES, 2015).

Revista Cubana de Ciencias Informáticas
Vol. 12, No. 1, Enero-Marzo, 2018
ISSN: 2227-1899 | RNPS: 2301
Pág. 43-57
http://rcci.uci.cu

Por otra parte, la empresa Sucuri ha reportado (Figura 1) que se han utilizado estas vulnerabilidades para comprometer entre el 8 y el 10% de las aplicaciones web atacadas, basadas en WordPress, en el 2016(SUCURI, 2016). Su utilización para la ejecución de ciberataques se ha manifestado del siguiente modo:

1. Descargar archivos arbitrariamente del servidor web, por ejemplo, el archivo wp-config.php con datos sensibles de administración:

http://victima.com/wp-admin/admin-ajax.php?action=revslider_show_image&img=../wp-config.php

2. Inclusión y modificación de archivos en el servidor. Esto se realiza mediante una petición HTTP POST y en el cuerpo se envían las instrucciones correspondientes. Como puede notarse, es bastante simple para un ciberatacante explotar estas vulnerabilidades. a través de rutas como:

http://victima.com/wp-admin/admin-ajax.php?action=revslider_ajax_action&client_action=get_captions_css

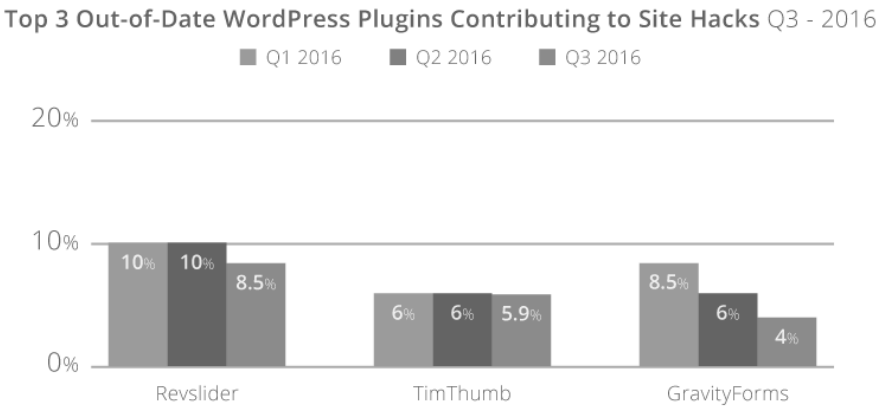

Figura 1. Utilización de complementos vulnerables para afectar aplicaciones web basadas en WordPress en el 2016. Fuente Website Hacked Trend Report 2016 - Q3 (Sucuri 2016)

A continuación, se expone el estudio sobre el comportamiento de los patrones de intentos de ciberataques asociados a la explotación de vulnerabilidades del componente RevSlider en los dominios y aplicaciones web de una entidad publicados en Internet.

Dominios explorados

Los intentos de ciberataques identificados a partir de los accesos estudiados fueron lanzados contra 22 dominios durante el período de 10 meses. Las peticiones no exploraron todos los dominios por meses. En la figura 2 puede

Revista Cubana de Ciencias Informáticas
Vol. 12, No. 1, Enero-Marzo, 2018
ISSN: 2227-1899 | RNPS: 2301
Pág. 43-57
http://rcci.uci.cu

apreciarse como los meses con menos dominios explorados fueron abril y mayo, mientras que los meses con más dominios explorados fue el período de octubre a diciembre.

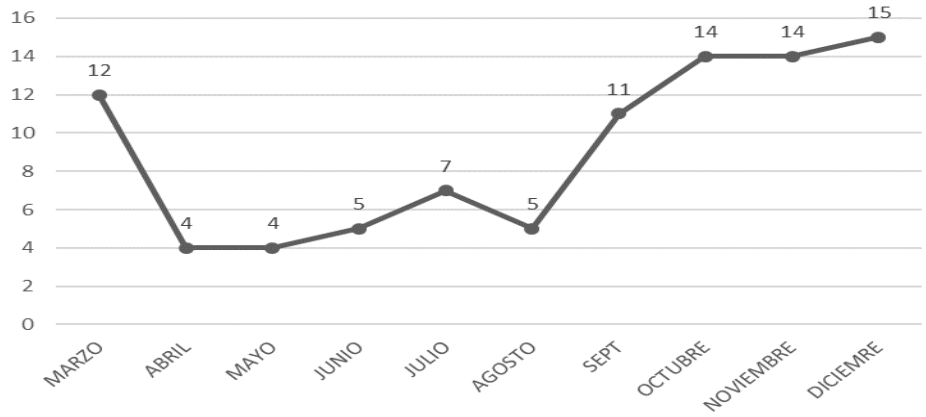

Figura 2. Cantidad de dominios explorados por meses.

Desde la perspectiva de los dominios, se registraron cinco (23%) que sufrieron intentos de ciberataques los 10 meses de estudio (figura 3). Es importante destacar que todos ellos estaban basados en WordPress. Sobre los restantes dominios explorados, solamente uno estaba basado en WordPress. Por tanto, podemos afirmar que se manifestó una tendencia de intentos de ciberataques periódicos contra dominios que representaban aplicaciones web basadas en esa tecnología.

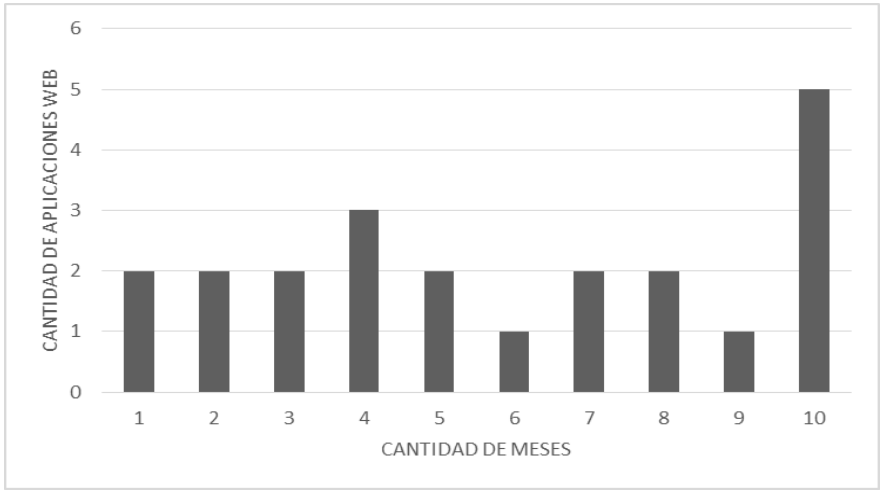

Figura 3. Cantidad de meses durante los cuales se exploraron los dominios únicos.

Rutas utilizadas

Los recursos fueron solicitados bajo siete patrones de ruta diferentes:

1. /wp-content/themes/ (…) /RevSlider/temp/update_extract/RevSlider/
2. /wp-content/plugins/ (…) /RevSlider/temp/update_extract/RevSlider/
3. /wp-content/plugins/ (…) /RevSlider/temp/update_extract/
4. //wp-content/plugins/RevSlider/
5. /wp-content/plugins/meteor-extras/includes/bundles/RevSlider/temp/update_extract/
6. /wp-admin/admin-ajax.php?
7. /

Los paréntesis de las rutas del uno al tres significan que en esas posiciones existían variaciones, como por ejemplo el nombre de los temas solicitados. La utilización de rutas con el prefjo wp (wp-content, wp-admin) permiten afirmar que el objetivo de los ciberatacantes eran aplicaciones web basadas en WordPress.

Temas utilizados

Fueron detectados 24 tipos o patrones de temas buscados en WordPress. Los temas sobre los cuales se concentraron los intentos de ciberataques fueron Avada, Centum y shoestrap. Estos temas se utilizaron en diferentes formas para construir los patrones rutas del uno al tres descritos en la sección anterior (Figura 5). Este comportamiento es semejante a otros descritos por empresas consultoras de seguridad en Internet (CID, 2014).

Revista Cubana de Ciencias Informáticas
Vol. 12, No. 1, Enero-Marzo, 2018
ISSN: 2227-1899 | RNPS: 2301
Pág. 43-57
http://rcci.uci.cu

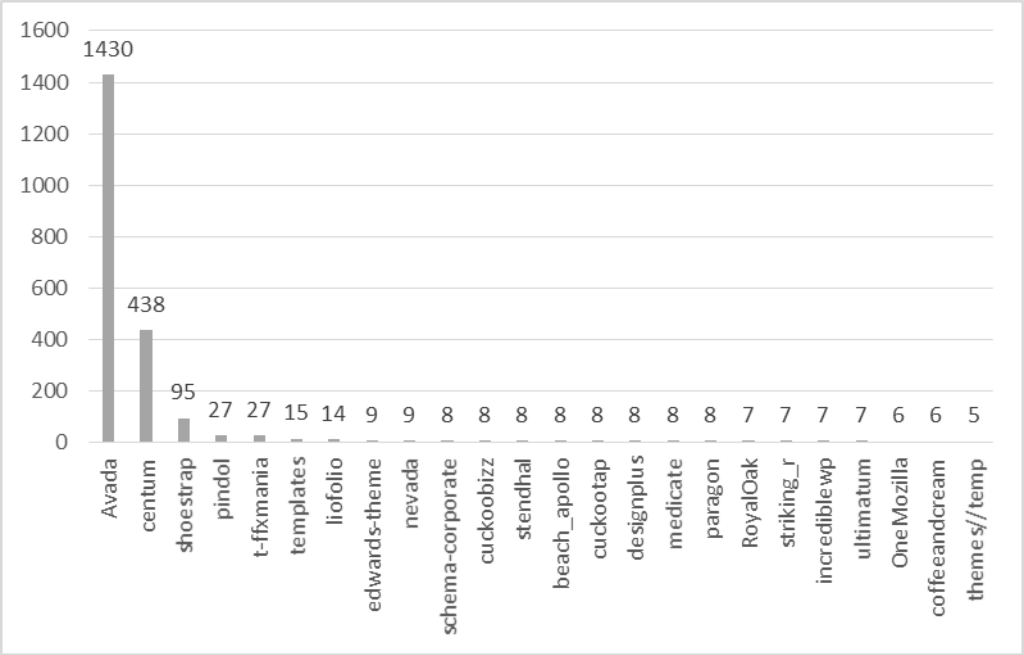

Figura 4. Peticiones de intentos de ataque por temas.

Nombres de recursos solicitados

El análisis de todas las peticiones de intentos de ataque estudiadas durante los 10 meses, mostró un total de 172 nombres de recursos únicos. En ningún mes se utilizaron todos los nombres de recursos. En la figura 5 se muestra como la utilización de los nombres varió entre un 16.9% y un 36.6% durante todos los meses estudiados.

Para lanzar el 85% del total de peticiones, se emplearon 16 nombres de recursos diferentes. De ellos, solamente ocho se utilizaron 10 meses, el resto fueron utilizados entre 4 y 9 meses. En la tabla 1 se muestra como los nombres de recursos más utilizados fueron .libs.php, myluph.php y joss.php. Esto es consistente con reportes en Internet registrados en variados portales de seguridad, realizados por administradores de varios países y en https://www.abuseipdb.com/.

Revista Cubana de Ciencias Informáticas
Vol. 12, No. 1, Enero-Marzo, 2018
ISSN: 2227-1899 | RNPS: 2301
Pág. 43-57
http://rcci.uci.cu

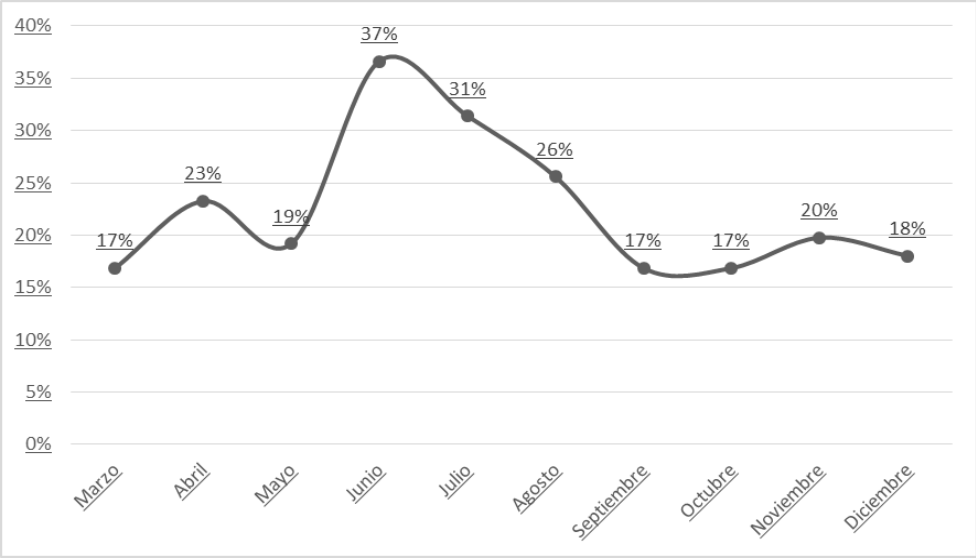

Figura 5. Utilización de nombres de recursos con respecto al total de nombres identificados.

Tabla 1. Nombres de recursos más utilizados.

Recurso	Porciento	Cantidad de meses utilizado
.libs.php	17%	10
myluph.php	17%	10
joss.php	9%	10
shunceng.php	7%	9
polahi.php	7%	9
me.php	7%	6
indra.php	5%	10
petx.php	3%	10
version.php	2%	9
db.php	2%	9
2x.php	2%	8
itil.php	1%	4
mil.php	1%	8
xxx.php	1%	10
love.php	1%	10
arhy.php	1%	10

Direcciones IP empleadas

Los intentos de ciberataques fueron lanzados desde 373 IP diferentes durante los 10 meses. El histograma de la figura 6 ilustra como la mayoría de las direcciones IP ejecutaron entre 1 y 46 peticiones maliciosas. El mes donde más

Revista Cubana de Ciencias Informáticas
Vol. 12, No. 1, Enero-Marzo, 2018
ISSN: 2227-1899 | RNPS: 2301
Pág. 43-57
http://rcci.uci.cu

direcciones IP se utilizaron fue julio con total de 78 y el mes que menos direcciones IP utilizaron fue diciembre con un total de 23 (figura 7).

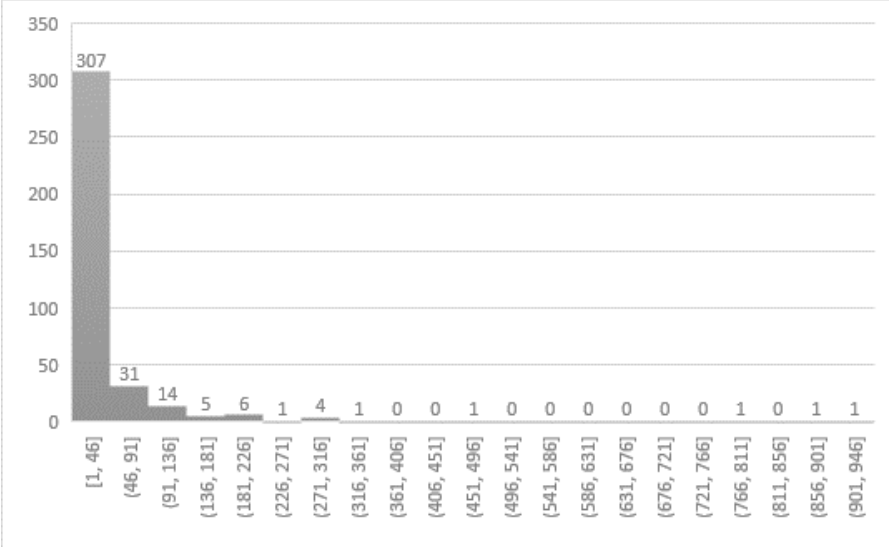

Figura 6. Histograma de los intentos de ciberataques realizados desde las direcciones IP.

Este comportamiento es típico de los procesos de reconocimiento e intentos de explotación de vulnerabilidades a través de botnets. Las botnets están conformadas por computadoras distribuidas a nivel global que comparten la característica de haber sido comprometidas por códigos dañinos o malware específicos de tipo persistentes, los cuales facilitan el control de estas para el lanzamiento de actividades maliciosas de forma remota, por lo que la ubicación geográfica de los ciberdelicuentes permanece en secreto, ya que los posibles afectados solo detectan la dirección IP de las computadoras controladas (BERTINO y ISLAM, 2017). Este aspecto es esencial para establecer medidas de detección y protección ante intrusos debido a que las direcciones IP no pueden ser bloqueadas por un tiempo indeterminado ya que con el transcurso del tiempo las máquinas afectadas son detectadas y descontaminadas y vuelven a realizar actividades legítimas (KIZZA, 2017).

Revista Cubana de Ciencias Informáticas
Vol. 12, No. 1, Enero-Marzo, 2018
ISSN: 2227-1899 | RNPS: 2301
Pág. 43-57
http://rcci.uci.cu

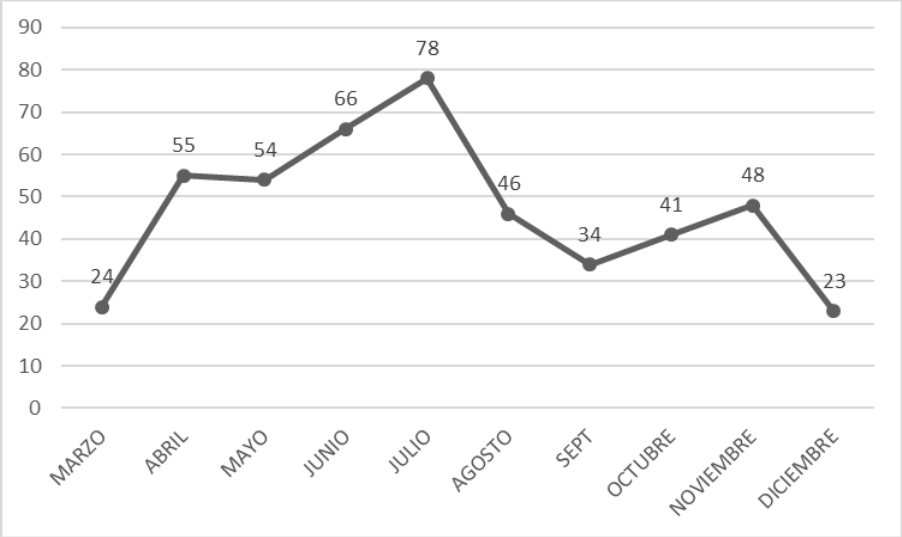

Figura 7. Número de direcciones IP empleadas por meses.

Mecanismos de protección

Las principales medidas de protección de las instalaciones de WordPress deben estar enfocadas a establecer procedimientos y medidas de mantenimiento y configuración que promuevan la solidez de las defensas ante estos intentos de ciberataques (DIEHL, 2016), por tanto:

- Deben utilizarse los complementos mínimos para garantizar la funcionalidad de la aplicación web. Estos a su vez deben de mantenerse actualizados y en el caso de que la última versión supere los tres meses, debe analizarse su sustitución por otro que cumpla la misma función ya que ello significa que el complemento quedó sin soporte y por tanto sin soluciones futuras de seguridad (CONȚU *et al.*, 2016).
- La medida anterior debe aplicarse a los temas empleados y aquellas que no se utilicen deben ser eliminados del servidor web.
- Deben deshabilitarse funciones del protocolo XML-RPC que sea vulnerables como system.multicall y pingback.ping (SILAEN y LIM, 2016).
- Deben interponerse firewall de aplicaciones web y sistemas de detección de intrusos enfocados no solo a la revisión de las peticiones recibidas sino también a la supervisión de los archivos de la instalación para alertar en caso de que sean modificados de algún modo por los ciberatacantes. No bastante en este punto solo tomar medidas para evitar una futura intrusión, sino que también es importante preparase para detectar a tiempo si esta tuvo lugar (PORAT *et al.*, 2017).

- Debe eliminarse siempre el usuario Admin y no emplear nombres de usuario que tengan relación con su función en la aplicación web como editor, autor, administrador u otro(MUSCAT, 2016).
- Utilizar contraseñas generadas por servicios y programas especializados para incrementar su fortaleza. Esto se aplica tanto a las contraseñas de administración como a la de conexión con la base de datos y otros servicios.
- Aplicar los permisos necesarios a los directorios del sistema operativo, así como las configuraciones propuestas por el servidor web (DE MEO y VIGANÒ, 2017).
- Cambiar el prefijo "wp_", de las tablas de la base de datos para que los códigos dañinos no puedan acceder a ellas si están programados para acceder a estas (SALLAM *et al.*, 2017).
- Limitar las direcciones IP mediante las cuales se tendrá acceso a las funcionalidades de administración y acceso de archivos de configuración.
- Instalar un componente de seguridad que limite los intentos de autenticación fallida (VAN ACKER *et al.*, 2017) como WP Limit Login Attempts.

Estas medidas pueden ampliarse según la disponibilidad de recursos computacionales y experiencia de administración, pero por si solas pueden garantizar un marco de trabajo de seguridad apropiado para el estado actual de la ciberseguridad en Internet.

Conclusiones

En el trabajo se describió el comportamiento de los intentos de explotación de las vulnerabilidades del componente RevSlider de WordPress. Mediante este estudio, los especialistas de ciberseguridad y desarrolladores de software, pueden conocer los mecanismos de reconocimiento y explotación sistemática que se ponen de manifiesto en Internet contra los dominios de aplicaciones web.

Pudo apreciarse como hubo una intensidad de intentos de ciberataques contra aplicaciones web basadas en WordPress. A pesar de que ninguna de estas aplicaciones poseía el componente RevSlider, los ciberataques se mantuvieron explorando la presencia o no de este.

Es importante destacar, además, cómo se continúan afectando aplicaciones web basadas en WordPress a través de la presencia de RevSlider tres años después de emitidas las alertas internacionales. La solución para evitar que estos ciberataques sean efectivos consiste en mantener la base tecnológica actualizada, no solamente el núcleo del CMS base con WordPress o Drupal, sino también los restantes componentes de la aplicación web. La utilización de complementos especializados de seguridad puede brindar una protección adicional importante.

Referencias

AKAMAI_TECHNOLOGIES. *Q1 2016 State of the Internet / Security Report* 2015

BERTINO, E. y ISLAM, N. Botnets and Internet of Things Security. *Computer*, 2017, vol. 50, nº 2, p. 76-79.

CARVER, J. C.; BURCHAM, M., *et al.* Establishing a baseline for measuring advancement in the science of security: an analysis of the 2015 IEEE security & privacy proceedings. En *Proceedings of the Symposium and Bootcamp on the Science of Security. 2016*. p. 38-51.

CID, D. *RevSlider Vulnerability Leads To Massive WordPress SoakSoak Compromise.* Sucuri Inc., 2014, vol. 20/11/2017, Disponible en: https://blog.sucuri.net/2014/09/slider-revolution-plugin-critical-vulnerability-being-exploited.html.

CONȚU, C. A.; POPOVICI, E. C., *et al.* Security issues in most popular content management systems. En *Communications (COMM), 2016 International Conference on (IEEE). Bucharest, Romania 2016*. p. 277-280.

CHIESA, R. y DE LUCA SAGGESE, M. Data Breaches, Data Leaks, Web Defacements: Why Secure Coding Is Important. En Ciancarini, P.;Sillitti, A.*et al* (editor). *Proceedings of 4th International Conference in Software Engineering for Defence Applications: SEDA 2015.* Cham: Springer International Publishing, 2016, p. 261-271.

DE MEO, F. y VIGANÒ, L. A Formal Approach to Exploiting Multi-stage Attacks Based on File-System Vulnerabilities of Web Applications. En Bodden, E.;Payer, M.*et al* (editor). *Engineering Secure Software and Systems: 9th International Symposium, ESSoS 2017.* Bonn, Germany: Springer International Publishing, 2017, p. 196-212.

DIEHL, E. Law 10: Security Is Not a Product, Security Is a Process. En *Ten Laws for Security.* Culver City, CA: Springer, 2016, p. 241-256.

HUANG, C.; LIU, J., *et al.* A study on Web security incidents in China by analyzing vulnerability disclosure platforms. *Computers & Security*, 2016, vol. 58, nº p. 47-62. ISSN 0167-4048.

JERKOVIĆ, H.; VRANEŠIĆ, P., *et al.* Securing web content and services in open source content management systems. En *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2016 39th International Convention on. 2016*. p. 1402-1407.

KAUR, R.; KAUR, A., *et al.* An Approach to Detect Vulnerabilities in Web-based Applications. *International Journal of Advanced Research in Computer Science*, 2017, vol. 7, nº 1, ISSN 0976-5697.

KENT, C.; TANNER, M., *et al.* How South African SMEs address cyber security: The case of web server logs and intrusion detection. En *Emerging Technologies and Innovative Business Practices for the Transformation of Societies (EmergiTech), IEEE International Conference on. 2016*. p. 100-105.

KIZZA, J. M. System Intrusion Detection and Prevention. En *Guide to Computer Network Security.* Cham: Springer International Publishing, 2017, p. 275-301.

LALLIE, H. S.; DEBATTISTA, K., *et al.* An Empirical Evaluation of the Effectiveness of Attack Graphs and Fault Trees in Cyber-Attack Perception. *IEEE Transactions on Information Forensics and Security*, 2017, vol. PP, nº 99, p. 1-1.

LE, V.-G.; NGUYEN, H.-T., *et al.* A Solution for Automatically Malicious Web Shell and Web Application Vulnerability Detection. En Nguyen, N.-T.;Iliadis, L.*et al* (editor). *Computational Collective Intelligence: 8th International Conference, ICCCI 2016. Proceedings, Part I*. Halkidiki, Greece: Springer International Publishing, 2016, p. 367-378.

MEDEIROS, I.; BEATRIZ, M., *et al.* Hacking the DBMS to Prevent Injection Attacks. En *Proceedings of the Sixth ACM Conference on Data and Application Security and Privacy. 2016*. p. 295-306.

MUSCAT, I. Web vulnerabilities: identifying patterns and remedies. *Network Security*, 2016, vol. 2016, nº 2, p. 5-10.

OKAMOTO, T. y TARAO, M. Toward an artificial immune server against cyber attacks. *Artificial Life and Robotics*, 2016, vol. 21, nº 3, p. 351-356. Disponible en: https://doi.org/10.1007/s10015-016-0282-9.

OWASP, T. Top 10-2013. *The Ten Most Critical Web Application Security Risks*, 2013, nº

PORAT, E.; TIKOCHINSKI, S., *et al.* Authorization Enforcement Detection. En *Proceedings of the 22nd ACM on Symposium on Access Control Models and Technologies. 2017*. p. 179-182.

RIADI, I. y ARISTIANTO, E. I. An Analysis of Vulnerability Web Against Attack Unrestricted Image File Upload. *Computer Engineering and Applications Journal*, 2016, vol. 5, nº 1, p. 19-28.

SALLAM, A.; BERTINO, E., *et al.* DBSAFE—An Anomaly Detection System to Protect Databases From Exfiltration Attempts. *IEEE Systems Journal*, 2017, vol. 11, nº 2, p. 483-493.

SILAEN, K. E. y LIM, C. A novel countermeasure to prevent XMLRPC WordPress attack. En *Data and Software Engineering (ICoDSE), 2016 International Conference on (IEEE). Denpasar, Indonesia 2016*. p. 1-6.

SINGHAL, A. y OU, X. Security Risk Analysis of Enterprise Networks Using Probabilistic Attack Graphs. En *Network Security Metrics*. Cham: Springer International Publishing, 2017, p. 53-73.

STAROV, O.; DAHSE, J., *et al.* No honor among thieves: A large-scale analysis of malicious web shells. En *Proceedings of the 25th International Conference on World Wide Web. Montréal, Québec, Canada. 2016*. p. 1021-1032.

SUCURI. *Website Hacked Trend Report 2016 - Q3*. Sucuri Inc. 2016

VAN ACKER, S.; HAUSKNECHT, D., *et al.* Measuring login webpage security. En *Proceedings of the Symposium on Applied Computing (ACM). Marrakech, Morocco 2017*. p. 1753-1760.

VASEK, M.; WADLEIGH, J., *et al.* Hacking is not random: a case-control study of webserver-compromise risk. *IEEE Transactions on Dependable and Secure Computing*, 2016, vol. 13, nº 2, p. 206-219.

VIANO, E. C. *Cybercrime, Organized Crime, and Societal Responses.* Springer International Publishing, 2017,

WANG, R.; ZHU, Y., *et al.* Detection of malicious web pages based on hybrid analysis. *Journal of Information Security and Applications*, 2017, vol. 35, nº p. 68-74.