Wordpress y los ataques XMLRPC

by javier - Lunes, abril 11, 2016

http://memoriasdeunsysadmin.info/wordpress-y-los-ataques-xmlrpc/

LA SEGURIDAD EN WORDPRESS

En este artículo voy a comentarles sobre uno de los ejemplos del por qué tu sitio o tu red, es tan seguro como el más débil de sus componentes. Wordpress no es excepción a la regla.

Hasta hace pocos meses atrás, pensaba que tenía cubierta la seguridad básica de mi blog y que al menos haría falta algo de dedicación para hackearlo. Bueno, una vez más, está demostrado que el "a mi no me va a pasar" es el peor error en el que uno puede caer.

Lo bueno es que siempre presto atención a los reportes y cuando comencé a recibir montones de reportes indicando bloqueos por intentos fallidos, decidí dar una mirada a los logs. Allí me llamó la atención una gran cantidad de registros de queries "POST" al /xmlrpc.php.

```
82.76.141.11 - - [20/Feb/2016:02:59:34 -0500] "POST /xmlrpc.php
HTTP/1.1" 200 415 "-" "-"
79.179.188.74 - - [20/Feb/2016:02:59:54 -0500] "POST /xmlrpc.php
HTTP/1.1" 200 415 "-" "-"
201.86.128.227 - - [20/Feb/2016:02:59:59 -0500] "POST
/xmlrpc.php HTTP/1.1" 200 415 "-" "-"
193.6.168.178 - - [20/Feb/2016:03:00:23 -0500] "POST /xmlrpc.php
HTTP/1.1" 200 415 "-" "-"
84.50.44.86 - - [20/Feb/2016:03:00:54 -0500] "POST /xmlrpc.php
HTTP/1.1" 200 415 "-" "-"
204.212.127.110 - - [20/Feb/2016:03:00:59 -0500] "POST
/xmlrpc.php HTTP/1.1" 200 415 "-" "-"
175.142.108.46 - - [20/Feb/2016:03:01:05 -0500] "POST
/xmlrpc.php HTTP/1.1" 200 415 "-" "-"
```

Logs del servidor web. Tengan en cuenta que cada línea, puede implicar miles de intentos de contraseñas, por lo que el impacto puede ser mucho mayor de lo que uno creería.

QUÉ ES XMLRPC?

Hasta entonces, un completo desconocido para mi, xmlrpc.php resultó ser un protagonista de una vulnerabilidad bastante sencilla de explotar.

Ahora bien, se preguntarán, como yo me pregunté, para qué sirve este bendito archivo?

XMLRPC es un protocolo que funciona como API y permite llamadas remotas. Algunos de los usos más

comunes de esta funcionalidad es el poder postear a través de un mail o un cliente de blogging; también es usado por varias funcionalidades de Jetpack(probablemente el plugin más útil y utilizado, a la vez, de wordpress); también permite comunicación con otros blogs.

Pero esto no es todo. Una de las llamadas que permite el protocolo, habilita a cualquiera a "multicalls", o sea, a realicar múltiples consultas en una sola vez. La aplicación más importante es el poder intentar miles de contraseñas en una vez, acelerando y amplificando los ataques de fuerza bruta. Un efecto secundario, pero no por eso menor, es el elevado consumo de recursos que provoca en el servidor, pudiendo llegar a convertirse en un DoS(denegación de servicio, al mantener saturado al servidor y suficientemente ocupado como para que no alcance a atender las peticiones de usuarios reales).

CÓMO PROTEGERSE CONTRA ATAQUES XMLRPC

La segunda pregunta que lógicamente se estarán haciendo. Bueno, existen muchas maneras, todas consisten en el bloqueo o restricción del uso del xmlrpc.php.

- ## XMLRPC ATTACKS BLOCKER

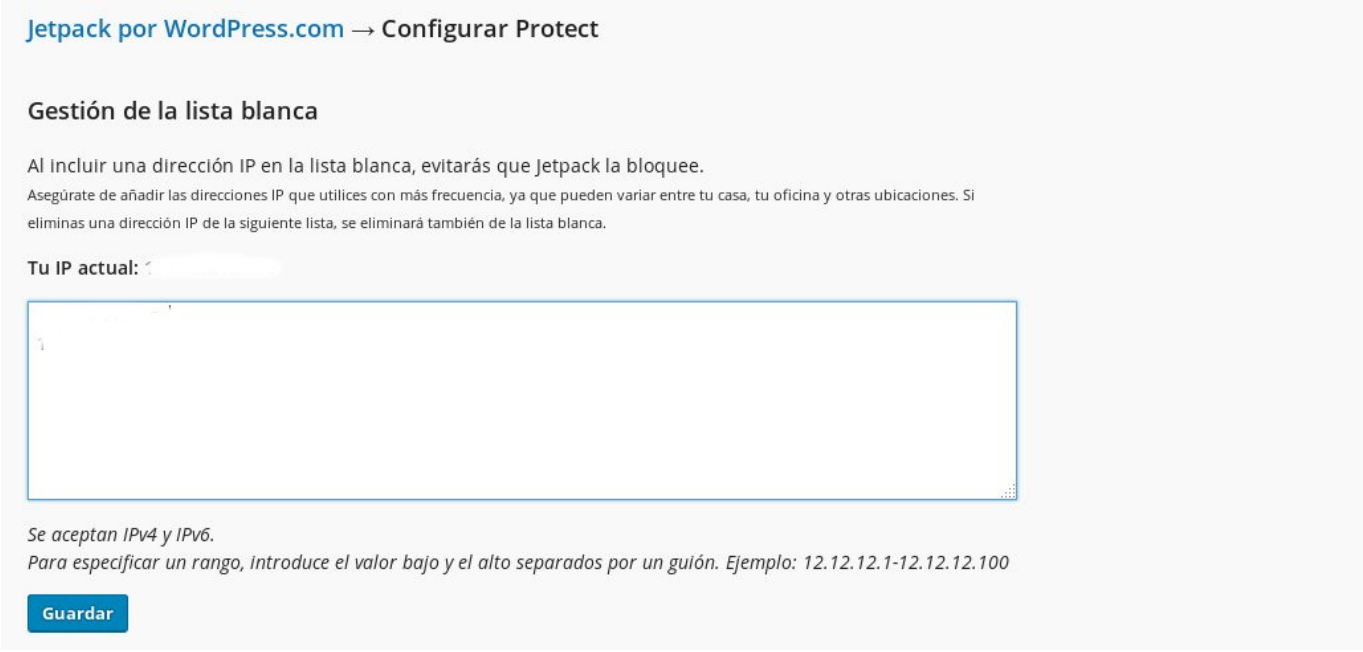
La mejor manera para esto, es simplemente, a través de "XMLRPC Attacks Blocker". Este plugin es gratuito y se puede descargar desde tu wordpress. El mismo nos da opciones muy útiles. Podemos, por ejemplo, permitir el acceso al archivo a un usuario único y bloquear automáticamente cualquier IP que intente acceder al mismo.

El inconveniente de este método, es que desactivar el uso de este archivo provocaría que Jetpack y algunas otras funcionalidades se rompan.

La gran ventaja, la sencillez, cualquiera puede usar esta opción. Si les preocupa esta vulnerabilidad, o bien si sospechan que están explotando esta vulnerabilidad contra su sitio, debieran considerar evaluar las funciones que realmente necesitas de Jetpack y buscar la manera de reemplazarlas, en lo posible

- ## JETPACK:

El mismo plugin permite proteger tu Wordpress de abuso sobre xmlrpc.php. Desde la función protect se puede habilitar fácilmente defensa muy efectiva contra diversos ataques de fuerza bruta. A la vez, dicha funcionalidad incluye una "lista blanca" para permitir acceso a IPs específicas.(Vale aclarar que, en Argentina al menos, un 90% de los usuarios seguramente tienen IP dinámica, con lo cual esto no les servirá de mucho...)



Protect restringe el acceso a xmlrpc, reduciendo la vulnerabilidad de tu Wordpress. Aquí vemos la whitelist de Protect.

- ## A TRAVÉS DE .HTACCESS:

Otra manera de protegerse consiste en bloquear el acceso a XMLRPC.PHP, a través del .htaccess. Para ello sigan los siguientes pasos:

- Edita el .htaccess, se encuentra en el raíz de tu sitio. Es posible hacerlo con el editor nativo que tienen los paneles de control, o bien pueden también descargarlo por ftp, editarlo con cualquier editor de texto y volver a subirlo. Si no sabés nada de código, es buena idea consultar a tu hosting o a un desarrollador. Inserta el siguiente código: RedirectMatch 403 /xmlrpc.php
- Del mismo modo, editen el archivo functions.php del tema que estén usando(Si en algún momento cambian de tema, no olviden este paso. Agreguen el siguiente código:
  function removeHeadLinks() {
  remove_action('wp_head', 'rsd_link');
  remove_action('wp_head', 'wlwmanifest_link');
  }
  add_action('init', 'removeHeadLinks');

- ## MANUAL DESDE APACHE:

Este método implica acceso SSH, con lo cual no es una opción para cualquiera, pero si es su caso y tienen algunos conocimientos de Apache, solo necesitan agregar lo siguiente dentro del <VirtualHost> del sitio en cuestión: <VirtualHost>
…

<files xmlrpc.php>
order allow,deny
deny from all
</files>
</VirtualHost>

Espero este artículo les sea útil. Tengan en cuenta finalmente que, en materia de seguridad, todo cambia todos los días, por lo que si conocen algún método fuera de los que hemos comentado, no duden en compartirlo!

REFERENCIAS:

Artículo de CloudFlare(En Inglés)

Artículo en Digital Ocean(En Inglés)

---

memoriasdeunsysadmin.tk