Analizando los logs de acceso de Apache

SERVIDORES

Entender y analizar los logs de acceso de Apache, nos puede resultar útil para cosas tan triviales como obtener estadísticas de visitas como para adoptar políticas de seguridad tanto preventivas como paliativas y en casos aún más graves, judiciales. En este artículo, haremos un breve resumen de como interpretar los logs de Apache y ver de qué forma nos pueden ser útiles.

Escrito por: **Eugenia Bahit** (Arquitecta GLAMP & Agile Coach)

Eugenia es **Arquitecta de Software**, **docente** instructora de tecnologías **GLAMP** (GNU/Linux, Apache, MySQL, Python y PHP) y **Agile coach** (UTN) especializada en Scrum y eXtreme Programming. Miembro de la **Free Software Foundation** e integrante del equipo de **Debian Hackers**.

Webs:
Cursos de programación a Distancia: www.cursosdeprogramacionadistancia.com
Agile Coaching: www.eugeniabahit.com

Redes sociales:
Twitter / Identi.ca: **@eugeniabahit**

La primera vez que observas el log de accesos de Apache, decides que lo mejor que podrías hacer con él es... es... en fin... cualquier cosa menos prestarle atención. Pero, *wait a minute little Saltamontes*, que no solo no es difícil entenderlos, sino que además, es mucho más útil de lo que te imaginas.

Los logs de Apache pueden configurarse mediante la directiva `CustomLog`[45] dentro del `VirtualHost` utilizando la sintaxis:

```
CustomLog /ruta/al/archivo formato
```

Donde formato podría ser `common` o `combined`. Por ejemplo:

45 http://httpd.apache.org/docs/2.2/mod/mod_log_config.html#customlog

```
CustomLog /miswebs/example.com/logs/access.log combined
```

La diferencia entre `common` y `combined` se podría decir que es poco sutil:

```
common:   %h %l %u %t \"%r\" %>s %b
combined: %h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-agent}i\"
```

Y es que el segundo, guarda la URI de referencia mediante la cuál se realizó la petición y el User-agent (información relativa al dispositivo utilizado para la conexión: navegador, buscador, etc.).

Los formatos anteriores pueden resultar incomprensibles, pero ¡no te estreses! Es mucho más sencillo de lo que imaginas:

```
%h      es el host que accede. Por ejemplo, una IP como:
        123.456.78.90

%l      el protocolo de identificación del usuario RFC 1413.
        no enloquezcas que seguramente verás un guión como salida a no
        ser que se opere en una red privada

%u      el nombre del usuario (comúnmente la salida será un guión a no
        ser que se trate de un usuario autenticado en el sistema)

%t      marca de tiempo: fecha completa incluyendo hora y UTC. Por ejemplo:
        10/Dec/2012:14:54:58 -0300

%r      (las barras que la envuelven son simples escapes de caracteres para
        que las comillas que le siguen, se impriman). Es la solicitud
        realizada por el usuario.
        Primero, incluye el método (GET, POST, PUT, etc.).
        Luego, el recurso solicitado (archivo al cuál se accede).
        Y finalmente, el protocolo (HTTP 1.1/1.0). Un ejemplo sería:
        "GET /index.php HTTP/1.1"

%>s     El código de respuesta de estado. Por ejemplo:
        200 (OK)
        404 (Not Found), etc.

%b      La cantidad de bytes entregados al usuario. Por ejemplo:
        4108
```

`combined`, suma además:

```
%{Referer}i      la URI de referencia (archivo, sitio o página que contiene
                 el vínculo hacia el recurso solicitado). Por ejemplo:
                 http://example.org/goTo.php?p=http://example.com/index.php

%{User-agent}i   El user-agent del usuario (valga la redundancia). Por ej.:
                 Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:11.0) Gecko/20100101 Firefox/11.0
```

A mi me gusta utilizar el formato `combined`. Es mucho más completo para obtener

estadísticas.

Por ejemplo, **el contador de descargas** que hice **para** la Web del Magazine **www.hdmagazine.org** es mitad un *script* de shell y mitad Python (ya que el mismo *script* de shell lo utilizo para otras cosillas xD), que básicamente lo que hace es un `grep` sobre los recursos coincidentes con una edición particular y los cuenta mediante la opción `-c`.

Algo tan simple como contar la cantidad de solicitudes realizadas al PDF de la edición anterior (edición número 1), se obtiene con solo 1 comando y 3 argumentos:

```
$ grep -c magazine=HackersAndDevelopers\&num=1\ HTTP access.log
```

Pero no solo es útil para obtener estadísticas de “visitas”. También nos puede servir para saber si algún usuario ha estado intentando acceder con intensiones “non santas”. Por ejemplo, es muy típico que alguien intente ingresar a alguna URL que contenga la palabra `admin`:

```
$ grep -i admin access.log | less
190.188.247.165 - - [03/Dec/2012:10:38:59 -0500] "GET
/admin/scripts/tinymce/jscripts/tiny_mce/plugins/ibrowser/scripts/phpThumb/phpThumb
.php?src=./index.php&fltr[]=blur|
5;echo+082119f75623eb7abd7bf357698ff66c>cache/acunetix; HTTP/1.1" 404 470 "-"
"Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:01 -0500] "GET /admin.htm HTTP/1.1" 404 429
"-" "Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:01 -0500] "GET /admin.html HTTP/1.1" 404 429
"-" "Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:02 -0500] "GET /admin HTTP/1.1" 404 426 "-"
"Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:02 -0500] "GET /admin/phpThumb/phpThumb.php?
src=./index.php&fltr[]=blur|5;echo+082119f75623eb7abd7bf357698ff66c>cache/acunetix;
HTTP/1.1" 404 438 "-" "Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64;
Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:02 -0500] "GET /Admin HTTP/1.1" 404 427 "-"
"Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:02 -0500] "GET /ADMIN HTTP/1.1" 404 426 "-"
"Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:02 -0500] "GET /admin/upload/phpThumb.php?
src=./index.php&fltr[]=blur|5;echo+082119f75623eb7abd7bf357698ff66c>cache/acunetix;
HTTP/1.1" 404 440 "-" "Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64;
Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:03 -0500] "GET /adminpanel HTTP/1.1" 404 430
"-" "Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:03 -0500] "GET /admin0 HTTP/1.1" 404 427 "-"
"Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:03 -0500] "GET /admin1 HTTP/1.1" 404 426 "-"
"Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:03 -0500] "GET /admin/release HTTP/1.1" 404
431 "-" "Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:03 -0500] "GET /admin_ HTTP/1.1" 404 427 "-"
"Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:04 -0500] "GET /_admin HTTP/1.1" 404 427 "-"
"Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:04 -0500] "GET
/admin/tiny_mce/plugins/ibrowser/scripts/phpThumb/phpThumb.php?
src=./index.php&fltr[]=blur|5;echo+082119f75623eb7abd7bf357698ff66c>cache/acunetix;
HTTP/1.1" 404 464 "-" "Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64;
Trident/5.0)"
```

```
190.188.247.165 - - [03/Dec/2012:10:39:04 -0500] "GET
/zadmin/tiny_mce/plugins/ibrowser/scripts/phpThumb/phpThumb.php?
src=./index.php&fltr[]=blur|5;echo+082119f75623eb7abd7bf357698ff66c>cache/acunetix;
HTTP/1.1" 404 465 "-" "Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64;
Trident/5.0)"
190.188.247.165 - - [03/Dec/2012:10:39:04 -0500] "GET /administrator HTTP/1.1" 404
432 "-" "Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)"
```

Y les aseguro que eso, es menos del 0,3% de las solicitudes que me encontré ese famoso 3 de diciembre, desde la misma IP **tras lanzarse la segunda edición de Hackers & Developers Magazine**.

La suma de los accesos realizados por la IP en cuestión, es una buena forma de obtener "el empujón final" para decidir bloquear a dicho usuario:

```
$ grep -c 190.188.247.165 access.log
7155
```

Parece una pavada ¿cierto? Pero no lo es.

Gracias a los logs de acceso de Apache, sabiendo analizarlos podemos tomar medidas preventivas importantes. Y no solo preventivas: en algunos países como la Rep. Argentina (donde Hackers & Developers Magazine tiene su asiento legal), este tipo de "pruebas" -según cada caso- podría ser considerada como un delito de ámbito penal. El "niñato" que anduvo *"jugando a ver si el software que me bajé de somosreguachihackers.hosting-gratuito.lala, hace algo cuando le doy clic al botón CheGuachinApretáAcá"*, ha dejado muchísimos rastros.

> ***En casos mucho más serios que este ejemplo, los datos que Apache provee, pueden ser utilizados a nivel forense*** *ya que rastrear una IP y obtener su localización aproximada no es difícil para los organismos de Justicia:* ***a través de la IP se puede obtener no solo la ubicación geográfica*** *(país, ciudad, provincia o estado), sino además, el proveedor ISP y la latitud y longitud del nodo.*

```
IP Address:  190.188.247.165          Location:  La Plata, Buenos Aires, Argentina
Coordinates: -34.9215, -57.9545       ISP:       Prima S.A.
```

Ejemplo de datos que pueden obtenerse tras un rastreo simple de la IP

Con las coordenadas del nodo, hasta Google Maps nos puede dar incluso, un domicilio bastante aproximado como se muestra en la siguiente imagen:

A nivel forense, estos datos son mucho más útiles que para nosotros: para la Justicia, solo bastará un oficio a la compañía proveedora del servicio, para obtener el nombre del cliente que utilizaba la IP en la fecha y hora especificada.

Es decir, que **gracias a los logs de Apache**, no solo podemos obtener divertidas estadísticas y tomar medidas preventivas, sino que además, **la Justicia puede obtener pruebas legalmente válidas**.

Y ahora ¿sigues pensando que revisar los logs de Apache es una pavada? A ti también te puede servir de mucho, puesto que no cabe duda de que **bloquear una IP en estos casos, no solo reducirá el ancho de banda consumido, sino que además, frenará un poco "la ansiedad" del niñato**:

```
iptables -I INPUT -s 190.188.247.165 -j DROP
```

Vale aclarar que **la revisión de los logs de acceso de Apache** -más allá de contar estadísticas-, debe ser asumida como **una política de seguridad a implementar de forma periódica**. Pues no existe un único usuario "molesto" ni tampoco los niñatos poseen IPs fijas.

La revisión de los logs de acceso de Apache, como política preventiva de seguridad, **SIEMPRE debería ir acompañada de la revisión de los logs de errores. Esto nos ayudará a asegurar aún más nuestras aplicaciones**, dado que, si haciendo un `grep` por la IP en cuestión sobre el log de errores, éste nos arroja algo, en el error (o los errores) arrojado estará la respuesta a la pregunta: ¿qué medida de seguridad se deberá sumar a nuestra aplicación?:

```
grep 190.188.247.165 error.log | less
```

Generalmente, la mayoría de los errores será del tipo 404 (archivo no encontrado):

```
... [error] [client 190.188.247.165] File does not exist:
/ruta/a/document/root/phpmyadmin
```

Pero **el punto de observación**, no **debe centrarse** en los errores 404, sino **en los errores de código** (500 y "compañía"). Son los únicos que nos ayudarán a saber qué parte del código de nuestra app, se debe asegurar. Una forma rápida de encontrar errores de PHP, por ejemplo, sería la siguiente:

```
grep 190.188.247.165 error.log | grep '\(Warning\|Notice\)' | less
```

*Como nota de color y a mero título personal, quiero admitir que **este tipo de "jueguecitos" resulta algo penoso cuando has pasado los últimos 16 años de tu vida, compartiendo con decenas de miles de personas** que no conoces, todo tu conocimiento **de forma absolutamente desinteresada**. Siempre digo que la vida no es ni justa ni injusta, sino que es "el resultado de las decisiones que uno toma". Y sentir que tu decisión de compartir con el mundo todo lo que sabes, trae aparejado que cualquier persona psicológicamente inestable intente dañarte, no es el sentimiento que elegiría para este fin de año.*

¡Gracias a quiénes saben valorar el esfuerzo! ¡Feliz 2013!

Eugenia Bahit
Responsable de Proyecto
Hackers & Developers Magazine

*Y por favor, ten siempre presente **"...que no son las cosas las que aportan significado a un momento determinado, sino que es el momento el que aporta significado a las cosas..."**.-*

*Frase pronunciada por el **Rabino Abraham Joshua Heschel***