Revista INGE CUC, Vol. 9, N° 2, pp 31-38, Diciembre, 2013

Seguridad en la configuración del Servidor Web Apache*

Security in the Apache Web Server Configuration

Artículo de Investigación Científica - Fecha de Recepción: 29 de Abril de 2013 - Fecha de Aceptación: 11 de Septiembre de 2013

Carlos Eduardo Gómez Montoya
Magíster en Ingeniería en Sistemas y Computación, Licenciado en Matemáticas y Computación, Universidad del Quindío. Armenia, Colombia. carloseg@grid.edu.co

Christian Andrés Candela Uribe
Magíster en Comercio Electrónico, Ingeniero de Sistemas y Computación, Universidad del Quindío. Armenia, Colombia christiancandela@grid.edu.co

Luis Eduardo Sepúlveda Rodríguez
Magíster en Software Libre, Ingeniero de Sistemas y Computación, Universidad del Quindío. Armenia, Colombia. lesepulveda@grid.edu.co

Para citar este artículo / To reference this article:
C. E. Gómez, C. A. Candela, and L. E. Sepúlveda, "Seguridad en la configuración del Servidor Web Apache," *INGE CUC*, vol. 9, no. 2, pp. 31–38, 2013.

Resumen: **Apache es el servidor Web con mayor presencia en el mercado mundial. Aunque su configuración es relativamente sencilla, fortalecer sus condiciones de seguridad implica entender y aplicar un conjunto de reglas generales conocidas, aceptadas y disponibles. Por otra parte, a pesar de ser un tema aparentemente resuelto, la seguridad en los servidores HTTP constituye un problema en aumento, y no todas las compañías lo toman en serio. Este artículo identifica y verifica un conjunto de buenas prácticas de seguridad informática aplicadas a la configuración de Apache. Para alcanzar los objetivos, y con el fin de garantizar un proceso adecuado, se eligió una metodología basada en el Círculo de Calidad de Deming, el cual comprende cuatro fases: planear, hacer, verificar y actuar, y su aplicación condujo el desarrollo del proyecto. Este artículo consta de cinco secciones: Introducción, Marco de referencia, Metodología, Resultados y discusión, y Conclusiones.**

Palabras clave: **seguridad en Apache, configuración de Apache, infraestructura de tecnología informática, servidor Web.**

Abstract: **Apache is the HTTP server or Web server with greater presence in the global market. Although its configuration is relatively simple, strengthening its security conditions involves understanding and applying a set of known, accepted and available general rules. Moreover, despite being an issue apparently resolved, the HTTP server security is a growing problem and not all companies take it seriously. This article entails identifying and verifying a set of computer security best practices applied to the Apache configuration. To achieve the desired objectives, and ensure an appropriate procedure, a methodology based on the Deming Quality Circle was chosen. It comprises four phases: plan, do, check and act. Their application led to the development of this project. This paper consists of five sections: Introduction, Framework, Methodology, Results and Discussion, and Conclusions.**

Keywords: **Apache security, Apache configuration, technology information infrastructure, Web server.**

* Artículo de Investigación científica derivado del proyecto de investigación titulado: Seguridad en la configuración del Servidor Web Apache. Universidad del Quindío. Fecha de Inicio: 1 de Julio de 2010. Fecha de Finalización: Febrero 28 de 2012.

I. Introducción

Cada vez son más frecuentes las noticias publicadas sobre incidentes de seguridad informática, tanto a nivel nacional como internacional. A pesar de esto, muchas organizaciones en Colombia apenas están empezando a invertir recursos económicos en este aspecto. Según los resultados de la *XI Encuesta Nacional de Seguridad Informática* [1], es notorio el avance de las organizaciones a nivel nacional frente a la gestión de incidentes de seguridad, sin embargo, muchos de estos esfuerzos no consiguen anticiparse a este tipo de actuaciones, resultando altamente perjudicadas. De igual forma, y en respuesta a la necesidad de cumplir con normatividad nacional y/o internacional, las empresas necesitan desarrollar planes de seguridad en la información, pero se encuentran con una limitada oferta en profesionales expertos en seguridad informática, por lo que deciden contratar a profesionales con poca experiencia.

Según [1], no menos del 40 % de los incidentes de seguridad ocurridos en Colombia durante 2010 están directamente relacionados con ataques a servidores Web. Los servidores orientan su operación sobre el protocolo HTTP. El protocolo de transferencia de hipertexto - HTTP es un protocolo de la capa de aplicación del modelo TCP/IP (*Transmission Control Protocol/Internet Protocol*) y es el corazón de la Web y define la forma como los clientes Web solicitan objetos a un servidor y cómo los servidores los transfieren a los clientes [11]. Los programas cliente HTTP son conocidos como navegadores de Internet, y entre los más populares se encuentran Mozilla Firefox [18] e Internet Explorer [13]. Por su parte, Apache Web Server [2] e Internet Information Server [14] son los servidores HTTP o servidores Web con mayor presencia en el mercado mundial [19].

Apache se caracteriza por ser estable, multiplataforma, modular y altamente configurable, lo cual significa que se puede adaptar para satisfacer diferentes necesidades. Apache registra en bitácoras (archivos *log*) los diferentes eventos que ocurren cuando está en servicio, y de esta manera facilita la obtención de estadísticas que son usadas para la toma de decisiones por parte del administrador. Además, dispone de componentes de seguridad, los cuales, si son configurados en forma apropiada, pueden ser aprovechados para fortalecer las condiciones de acceso a los recursos Web disponibles para ser recuperados a través de solicitudes HTTP realizadas por un navegador. Apache se caracteriza también por ser de código abierto (*Open Source*) y gratuito [23].

Poner en operación un Servidor Web Apache para una organización requiere tener en cuenta muchas variables, entre las que se destacan la ubicación física del *hardware* que soporta el sitio Web; las condiciones de seguridad física y disponibilidad eléctrica; la persona que administra el Servidor y sus conocimientos, capacidad y experiencia; y las condiciones del servicio de alojamiento de los documentos.

La configuración de Apache se realiza mediante la edición del archivo de texto *httpd.conf*, el cual tiene todas las instrucciones que debe seguir Apache para su funcionamiento. Aunque la configuración básica de Apache es un procedimiento relativamente sencillo, para ajustar la configuración y fortalecer sus condiciones de seguridad es necesario entender y aplicar un conjunto de reglas generales ya conocidas, aceptadas y disponibles [8]. Sin embargo, a pesar de ser un tema aparentemente resuelto, la seguridad en los servidores HTTP constituye un problema en aumento, y no todas las compañías que tienen publicada su información o servicios Web disponibles para los usuarios toman en serio esta problemática. Solo cuando reciben ataques a su seguridad intentan tomar medidas al respecto, muchas veces sin denunciar ante las autoridades por temor a perder prestigio ante sus clientes. Es así como [1] muestra que en 2010 el 21,5 % de las organizaciones en Colombia manifestó que ha tenido accesos no autorizados a sus sitios Web y el 13,08 % ha tenido ataques de negación de servicio que ha afectado la disponibilidad de su información publicada en Internet.

La informática ha traído grandes beneficios a la humanidad, pero también ha permitido la aparición de nuevos peligros relacionados con la seguridad de la información. Esta situación es un fenómeno de impacto mundial, y se evidencia en la utilización de la información con fines delictivos, tales como: la suplantación de identidad, la captura y tráfico de información e incluso el robo electrónico de dinero. Todo esto debido al desarrollo de nuevas tecnologías de la información y la comunicación que permiten gestionar servicios informáticos en los cuales se debe almacenar, transmitir y procesar información a través de redes distribuidas geográficamente. La seguridad de la información cada vez más se está convirtiendo en un área crítica en todo tipo de organizaciones. Existe información confidencial que debe ser mantenida como tal para que no llegue a manos equivocadas; existe también información pública que no debe ser modificada sin autorización o por equivocación; y se debe permitir el acceso a la información en forma oportuna y sin interrupciones para los usuarios autorizados. Estos elementos, la confidencialidad, la integridad y la disponibilidad, son los elementos fundamentales de la seguridad de la información [11]. Otros aspectos importantes de la seguridad de la información son la autenticación, el no repudio y el control de acceso [24].

Obtener información sobre los incidentes de seguridad que han afectado las organizaciones y tratar de aprender de ellos no es fácil, porque una porción significativa de los afectados normalmente no está interesada en divulgar estos eventos debido al perjuicio que significa para sus negocios [1]. Por esta razón, muchas organizaciones inician por separado el ejercicio de tratar de mitigar las vulnerabilidades que conocen puntualmente y, por lo general, dejan una gran brecha por cubrir; otras, por el contrario,

no hacen el menor esfuerzo por considerar el tema de seguridad dentro de la agenda de la organización, además de contar con presupuestos limitados, y muy a menudo con una férrea indiferencia por parte de las directivas de las organizaciones.

De acuerdo con [8], es clave tener en cuenta las siguientes recomendaciones de seguridad: cifrar la información, especialmente las contraseñas almacenadas; eliminar paquetes, aplicaciones, componentes o módulos innecesarios, debido a que cada herramienta instalada puede convertirse en un agujero de seguridad; hacer una adecuada gestión de usuarios y grupos, identificando los privilegios y restricciones de cada uno de ellos; mantener actualizado el sistema y estar atento a las novedades de seguridad que publican los fabricantes; utilizar herramientas de seguridad adicionales, como un *firewall* que controle las conexiones abiertas y pueda detectar actividades sospechosas; y revisar el contenido de los archivos *log*.

Conocer las características del Servidor Web Apache en materia de seguridad, aprender a utilizarlas e identificar las vulnerabilidades más significativas de este y establecer las contramedidas, permitirán aumentar el nivel de seguridad en condiciones aceptables para las organizaciones. Una configuración adecuada de Apache permite, además, evitar que se entregue información en las líneas de encabezado de los mensajes de respuesta HTTP, que pueda ser utilizada por un atacante; y bloquear el acceso de los usuarios remotos a los directorios que están fuera de la estructura de documentos ofrecidos por el servidor. También se puede configurar Apache para que reciba conexiones seguras mediante el protocolo HTTPS, es decir, HTTP sobre SSL (*Secure Socket Layer*) [8].

Adicionalmente, es conveniente instalar y configurar el *firewall* de aplicación ModSecurity [16], el cual consiste en una herramienta de registro, detección y mitigación contra ataques al Servidor Web Apache [8].

El proyecto de investigación *Seguridad en la configuración del Servidor Web Apache* fue realizado con el fin de estudiar las principales características de Apache Web Server en materia de seguridad, analizar vulnerabilidades e identificar y comprobar un conjunto de recomendaciones y buenas prácticas para mitigar los problemas de seguridad identificados a partir de su configuración.

Con el fin de alcanzar estos objetivos, y para garantizar un proceso adecuado se eligió una metodología basada en un esquema de calidad ampliamente aceptada como es la aplicación del Círculo de Calidad de Deming [17].

Según [22], la calidad representa un proceso de mejora continua, en el cual todas las áreas de una organización buscan satisfacer las necesidades del cliente o anticiparse a ellas, participando activamente en el desarrollo de productos o en la prestación de servicios.

Por otra parte, la calidad depende de la manera en la que se organizan las actividades requeridas para la fabricación de un producto o la prestación de un servicio. Para obtener un acercamiento a la calidad es recomendable la utilización de una metodología aceptada que facilite este proceso a una organización. El Círculo de Calidad de Deming determina un modelo simple y eficaz para controlar la calidad a partir de cuatro pasos que se deben seguir: planear, hacer, verificar y actuar [17].

El resto del artículo está estructurado de la siguiente manera: la sección II presenta el marco de referencia; en la sección III se describe la metodología empleada en el desarrollo del proyecto; los resultados y discusión se presentan en la sección IV y las conclusiones en la sección V.

II. Marco De Referencia

A. Seguridad Informática

En un principio, la seguridad en redes de computadoras no era considerada, porque el acceso a ellas era restringido y las aplicaciones eran principalmente educativas y de investigación [24]. Sin embargo, con el paso del tiempo, cada vez más personas hacen uso de aplicaciones en las que los datos viajan por la red, y en las cuales la seguridad es un aspecto crítico que debe ser analizado desde diferentes puntos de vista para tratar de hacer más seguras las aplicaciones.

La seguridad es un área de enormes proporciones: está relacionada con la confidencialidad de los datos, la integridad, el control de acceso no autorizado, la verificación de identidad de quien origina un mensaje y con la disponibilidad de la información, entre muchos otros.

Los problemas de seguridad generalmente son generados por personas malintencionadas que intentan obtener beneficios o hacer daño a otros. Estas personas comúnmente tienen recursos técnicos y económicos, y están dedicadas a buscar la forma de aprovechar vulnerabilidades, por lo que es necesario estar preparados y atender con seriedad las recomendaciones de los expertos.

Los problemas de seguridad se pueden clasificar fundamentalmente en: confidencialidad, integridad y disponibilidad. La *confidencialidad* consiste en mantener en secreto información valiosa con el fin de prevenir que usuarios no autorizados puedan acceder a ella. La *integridad* intenta prevenir: la modificación de la información por usuarios no autorizados, la modificación no autorizada o no intencionada de usuarios autorizados y la preservación de la consistencia de la información. La *disponibilidad* asegura que los usuarios autorizados de un sistema tengan acceso a la información en el sistema y a la red en forma oportuna y sin interrupciones. Dependiendo de la aplicación y del contexto, uno de estos principios podría ser más importante que los otros.

Otros aspectos importantes son la autenticación, la autorización y el no repudio. La *autenticación* se encarga de validar la identidad del interlocutor antes de revelar información sensible. La *autorización* es el conjunto de privilegios que tiene un usuario de un sistema, lo que determina las acciones y el comportamiento para usuarios individuales o por categorías. El *no repudio* permite comprobar que un mensaje fue realmente emitido por quien lo firma [24].

Una parte importante de la seguridad en las redes de computadoras está basada en la criptografía, la cual ofrece los servicios de confidencialidad, autenticación y control de integridad. La criptografía ofrece una solución frente a estas necesidades que son fundamentales en las transacciones electrónicas del mundo de hoy.

El cifrado es una herramienta para proteger secretos. Es posible cifrar archivos almacenados en el disco duro de una computadora para evitar que el robo o la pérdida de la computadora pudieran comprometer los datos del propietario [6]. También es posible cifrar los datos que se envían a través de la red de computadoras, especialmente los que se envían a una entidad financiera o información relacionada con antecedentes médicos de una persona.

Los sistemas criptográficos tradicionales han sido divididos históricamente en dos categorías: *cifrados por sustitución*, donde cada símbolo del lenguaje con el que se escriben los mensajes es reemplazado por otro; y *cifrados por transposición*, donde se conservan los símbolos pero se les cambia el orden en una forma sistemática a partir de una clave que determina el procedimiento realizado [24].

Con la llegada de las computadoras se dio paso a la criptografía moderna, dado que permitía superar las limitaciones de la criptografía tradicional en relación con volumen de operaciones por realizar [24].

La criptografía moderna usa las mismas ideas básicas de la criptografía tradicional (la sustitución y la transposición), pero su orientación es distinta. En la actualidad se utilizan algoritmos mucho más sofisticados, los cuales son muy difíciles de aplicar usando procedimientos manuales [6].

Un sistema criptográfico es usado para cifrar o descifrar datos. Los sistemas criptográficos modernos vienen en diferentes formas: *sistema criptográfico de llave simétrica,* el cual usa una llave simétrica tanto para cifrar como para descifrar datos; *sistema criptográfico de llave pública* o *sistema criptográfico de llave asimétrica*, en el que se usa un par de llaves: una pública, que puede ser libremente distribuida, y otra privada, que debe mantenerse en secreto; y un *sistema criptográfico híbrido* combinando los dos anteriores que se utiliza con el fin de intercambiar una llave simétrica (usualmente de sesión) entre dos usuarios que están conectados a través de una red de computadoras.

Un *message digest* (MD) es un método que proporciona control de integridad. Este método calcula un resumen de un mensaje (también llamado "huella digital") a través de una función de *hashing*, la cual toma un mensaje y a partir de él calcula un número de longitud fija. Una función de *hashing* tiene las siguientes propiedades [11]:

- Dado el mensaje m, es fácil calcular MD (m).
- Dado MD (m), no es posible encontrar el mensaje m que fue utilizado para calcularlo.
- Dado un mensaje m, no es posible encontrar otro mensaje m' tal que produzcan el mismo MD.
- Cualquier cambio en el mensaje de entrada, aunque sea de un solo bit, produce un MD distinto.

Un *digest* (MD) es usado con el fin de determinar si un mensaje ha sido alterado o no después de haberse producido. Es decir, si un documento ha sido modificado, el *digest* va a ser diferente.

Los dos métodos más conocidos para calcular un *digest* son MD5 (*Message Digest Version 5*) y SHA-1 (*Secure Hash Algorithm Version 1*), que producen MDs de 128 y 160 bits, respectivamente.

Calcular el *digest* de un mensaje es utilizado para realizar el control de integridad; y si no es necesaria la confidencialidad, ahorra tiempo, tanto de cifrado como de transmisión, porque lo que usualmente se cifra (con llave privada) es el *digest* del mensaje (y no el mensaje), debido a que un *digest* es de longitud limitada, mientras que el mensaje no.

Por otra parte, existe el HMAC (*Keyed-Hash Message Authentication Code*), el cual realiza el cálculo de un *digest* mediante una función de *hashing* como MD5 o SHA-1, junto con una clave secreta. Este cálculo proporciona tanto integridad como autenticación.

La criptografía de llave pública hace posible que las personas que no comparten una llave común se puedan comunicar con seguridad. También permiten firmar mensajes sin la presencia de un tercero que pueda certificar su autenticidad. Sin embargo, hay un inconveniente: cuando A y B no se conocen, ¿cómo puede cada uno obtener la clave pública del otro para poder comunicarse? Es necesario un mecanismo para asegurar que las claves públicas puedan intercambiarse de manera segura [24].

Un *certificado digital* es un documento digital mediante el cual un tercero confiable o autoridad certificadora garantiza la vinculación entre la identidad de un sujeto o entidad y su llave pública.

Si bien existen variados formatos para certificados digitales, los más comúnmente empleados se rigen por el estándar X.509. El certificado contiene usualmente el nombre de la entidad certificada, un número de serie, la fecha de expiración, una copia de la llave pública del titular del certificado (utilizada para la verificación de su firma digital) y la firma digital de la autoridad emisora del certificado, de forma que el receptor pueda verificar que esta última ha establecido realmente la asociación. La en-

tidad certificadora debe proteger muy bien su llave privada.

Un certificado digital es usado para difundir la llave pública de su titular; este certificado se obtiene mediante una solicitud a una entidad certificadora. Un certificado digital puede entregarse en un medio físico (tarjeta) para mayor seguridad, aunque es un procedimiento no escalable.

Cada vez que un transmisor desea verificar la identidad del receptor, primero solicita su certificado digital. A partir de este puede obtener su llave pública, verificar su identidad y enviarle información de manera segura.

Algunas de las entidades certificadoras más conocidas son Verisign [25], MCI (Microwave Communications, Inc.) [12], AT&T (American Telephone and Telegraph Company) [3], y en Colombia Certicámara [5]. Los certificados digitales tienen validez legal en Colombia siempre y cuando sean expedidos por una entidad certificadora legalmente reconocida en el país.

B. Protocolo HTTP

Hasta comienzos de la década de 1990 Internet fue usada principalmente por investigadores, académicos y estudiantes universitarios para entrar en *hosts* remotos, para transferir archivos de un *host* local a un *host* remoto y viceversa y para recibir y enviar correo electrónico. Aunque esas aplicaciones fueron (y continúan siendo) extremadamente útiles, Internet fue esencialmente desconocida fuera de la comunidad académica y de investigación [11]. Luego, a comienzos de la década de 1990, surgió una nueva aplicación, la World Wide Web, una aplicación de Internet que permite enlazar información, representada generalmente en forma de páginas Web, las cuales pueden contener texto, gráficas, animaciones, audio, video e hipervínculos. Los enlaces dentro de las páginas Web permiten conectar una página Web con otros recursos, bien sea localmente o en servidores remotos, sin necesidad de conocer la real ubicación del recurso [10].

Una página Web (también llamada documento Web) se compone de objetos. Un *objeto* es simplemente un archivo, que puede ser HTML, una imagen JPEG, un applet de Java o un clip de video. Todo objeto Web es alcanzable mediante un único URL [11]. Un URL (*Uniform Resourse Locator*: Localizador Uniforme de Recursos) es la forma más común de identificar un recurso Web. Un URL describe la ubicación específica de un objeto en un servidor Web particular, indicando de manera precisa y sin ambigüedad el nombre del *host*, el número del puerto y la ruta completa del objeto en ese servidor [9].

El principal atractivo para la mayoría de los usuarios consiste en que la Web permite que los usuarios reciban las páginas y otros recursos Web en el momento que lo deseen. Además, los hipervínculos y motores de búsqueda ayudan a los usuarios a navegar a través de un enorme conjunto de sitios Web [11].

HTTP (*HyperText Transfer Protocol*: Protocolo de Transferencia de Hipertexto) es un protocolo de la capa de aplicación del modelo TCP/IP y es el corazón de la Web. Está definido en los RFC 1945 [4] y 2616 [7]. HTTP es implementado por dos tipos de programas: un programa cliente y un programa servidor. El programa cliente y el programa servidor son ejecutados en diferentes *hosts* y se comunican entre sí mediante el intercambio de mensajes HTTP. El protocolo HTTP define la estructura de esos mensajes y la forma como son intercambiados por el cliente y el servidor [11].

C. Servidor Web Apache

Apache es un extraordinario servidor Web (servidor para el protocolo HTTP) distribuido por [2]. De acuerdo con [19], Apache tiene una participación superior al 60 % de los servidores en todo el mundo.

Apache se caracteriza por ser estable, multiplataforma, modular y altamente configurable, lo cual significa que se puede adaptar para satisfacer diferentes necesidades. Apache registra los diferentes eventos que ocurren cuando está en servicio a través de archivos *log*. De esta manera facilita la obtención de estadísticas que son usadas para la toma de decisiones por parte del administrador. Además, dispone de componentes de seguridad, los cuales pueden ser aprovechados para fortalecer las condiciones de acceso a recursos Web disponibles para ser recuperados a través de solicitudes HTTP realizadas por un navegador, siempre y cuando sean configurados apropiadamente. Apache se caracteriza también por ser *Open Source* y gratuito [23].

La configuración de Apache se realiza mediante la edición del archivo de texto *httpd.conf*, el cual tiene todas las instrucciones que debe seguir Apache para su funcionamiento.

III. Metodología

La metodología empleada en el desarrollo de este proyecto obedece a un esquema de calidad basado en el Círculo de Calidad de Deming [17] y consta de cuatro fases: levantamiento de información; análisis de información e identificación de vulnerabilidades; investigación de contramedidas; y realización de pruebas. A continuación se explica cada una de estas fases.

La primera fase, *levantamiento de información*, incluye un estudio teórico sobre el protocolo HTTP, el Servidor Web Apache y un panorama general de la seguridad de la información a nivel general y, específicamente, sobre la publicación de contenidos en Internet.

En la segunda fase, *análisis de información e identificación de vulnerabilidades*, se realizó un estudio de los marcos de referencia asociados a la seguridad de la información y se analizaron los resultados de la *XI Encuesta Nacional de Seguridad Informática*, en la cual se consultó a los responsables de los sitios Web de algunas de las más importantes empresas del país sobre las costumbres y precauciones que tienen en materia de seguridad. Posteriormente se identificaron herramientas para detectar vulnerabilidades en servidores Web basados en Apache.

La tercera fase, *investigación de contramedidas*, se enfocó en la recopilación de recomendaciones y buenas prácticas de seguridad informática en general y sobre la instalación y configuración del Servidor Web Apache en particular.

La última fase consistió en la implementación de un laboratorio virtual usando *software* de virtualización de sistemas operativos para el montaje de los servidores Windows Server y GNU/Linux, con el fin de recrear los escenarios y hacer posible la utilización de las herramientas y la verificación de las mismas mediante pruebas funcionales.

Como lo recomienda [17], el esquema de calidad es un proceso cíclico que se debe realizar constantemente para asegurar la correcta organización de las actividades en un proceso determinado para cualquier tipo de organización.

IV. Resultados y Discusión

De acuerdo con la metodología utilizada, y luego de realizar los estudios teóricos previos, y analizar las vulnerabilidades del Servidor Web Apache, se procedió a realizar pruebas funcionales en el laboratorio en las que se construyeron escenarios virtuales para recrear el problema objeto de estudio con el fin de identificar las contramedidas adecuadas para las vulnerabilidades evidenciadas durante las pruebas realizadas y comprobar una configuración más segura del Servidor Web Apache.

Como lo sugiere [15], la realización de pruebas de penetración permite obtener las vulnerabilidades conocidas de un sistema. Existen diversas herramientas para realizar este tipo de pruebas, entre las que se pueden mencionar *Nessus*, *MetaExploit*, *Saint*, *Sara* y *Nikto*.

En el caso particular de este proyecto, las pruebas realizadas se enfocaron hacia Nikto, por ser un escáner de vulnerabilidades que se concentra exclusivamente en servidores Web. El escáner, luego de realizar un análisis a un sitio Web, da como resultado un informe detallado sobre las vulnerabilidades encontradas. La información puede ser utilizada para tomar medidas de precaución para prevenir posibles ataques al sitio Web [20].

El escáner busca principalmente fallas en la configuración de un servidor Web; problemas derivados de la ejecución de archivos y *scripts* automáticamente o que se han identificado como inseguros; o los problemas que ocasionan las versiones desactualizadas de *software* [20].

Dada la importancia de este tipo de herramientas para realizar análisis de vulnerabilidades, es recomendable que dichas herramientas cuenten con la versión actualizada de las vulnerabilidades conocidas para no obtener resultados confusos e imprecisos.

Inicialmente se realizaron instalaciones del Servidor Web Apache en los sistemas operativos GNU/Linux y Windows Server, usando configuraciones por defecto, con el fin de estudiar diferentes formas de instalación e identificar vulnerabilidades derivadas de este proceso. Posteriormente se realizaron pruebas de penetración mediante Nikto mediante el cual se obtuvieron resultados similares en los dos sistemas operativos, en los que la vulnerabilidad más relevante detectada en este proceso fue la utilización de una versión desactualizada del Servidor Web, como se puede apreciar en la figura 1.

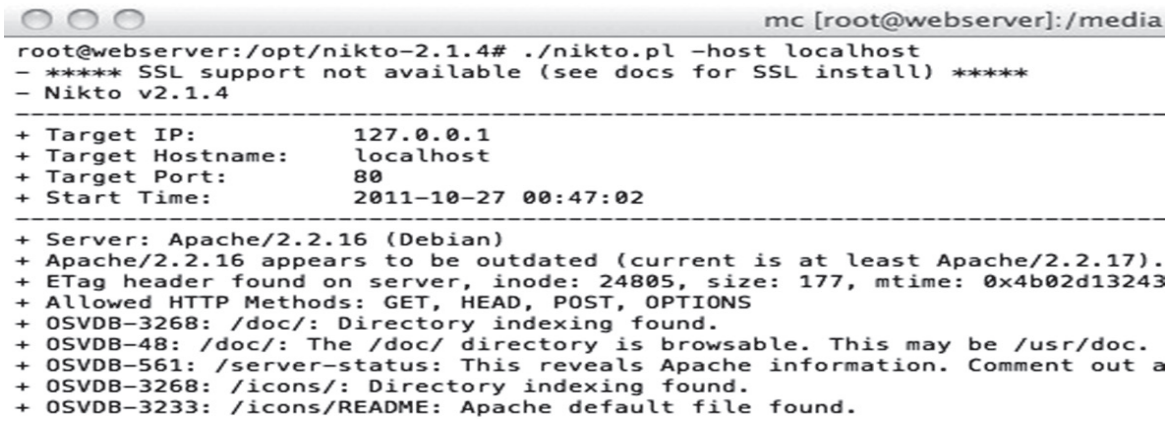

Figura 1. Resultado de la ejecución de Nikto

Posteriormente se procedió a instalar una versión actualizada del sistema, obtenida del sitio Web de Apache [2], y se realizó nuevamente una prueba de penetración, en la que las vulnerabilidades más relevantes fueron la aceptación de la línea de encabezado ETag para validar contenido almacenado en caché y aceptación de solicitudes HTTP con el método *Trace*. Se debe destacar que lo más importante al realizar estos análisis de vulnerabilidades es que luego de la realización de una misma prueba puede dar resultados diferentes si son realizadas en diferentes tiempos.

Luego de obtener las vulnerabilidades se estudió cada una de ellas, se establecieron las contramedidas correspondientes para cada caso y se aplicaron a la configuración del Servidor Web Apache en los escenarios virtuales. Es importante resaltar que al intentar contrarrestar una vulnerabilidad (por ejemplo, la versión desactualizada del Servidor Web Apache) pueden surgir nuevas vulnerabilidades, lo que hace necesario repetir el procedimiento para comprobar la efectividad de las contramedidas aplicadas e identificar nuevas vulnerabilidades derivadas del proceso realizado.

Además de los hallazgos realizados por las herramientas tipo escáner aplicadas al Servidor Web Apache existe un conjunto de acciones recomendables para mejorar las condiciones de seguridad relacionadas con la configuración del Servidor Web Apache. Entre las recomendaciones o buenas prácticas más relevantes están las siguientes: tener un conocimiento completo de todos y cada uno de los elementos del archivo de configuración de Apache (estructura, parámetros y valores), ya que incluir elementos desconocidos o cargar módulos innecesarios puede comprometer la seguridad del sistema; realizar una gestión de usuarios de manera que se garantice el monitoreo constante del ciclo de vida de los usuarios del sistema; esto incluye eliminar cuentas de usuario innecesarias para la prestación del servicio y analizar el alcance de permisos y privilegios de cada usuario del sistema; verificar constantemente la existencia de archivos en el sistema con permisos de ejecución y monitorear el listado de tareas programadas para evitar la ejecución de acciones no previstas por el administrador del sistema; configurar la huella identificativa del Servidor Web con la mínima información necesaria e incluso, dependiendo de la necesidad, utilizar esta huella identificativa como un elemento distractor para posibles atacantes, cambiándola para indicar un tipo de Servidor Web diferente; y una recomendación final: desplegar los servicios correspondientes al Servidor Web en ambientes operativos exclusivos con el ánimo de evitar vulnerabilidades expuestas por otros servicios prestados en el mismo sistema [15], [21].

La aplicación de las recomendaciones o buenas prácticas mencionadas, junto con la realización de análisis de vulnerabilidades, no son suficientes para fortalecer las condiciones de seguridad de un Servidor Web Apache. De acuerdo con [15] y [21], es conveniente complementar las medidas de seguridad con la instalación y configuración de un *firewall* de nivel de aplicación. *ModSecurity* [16] es el firewall de aplicación más comúnmente utilizado para aumentar las capacidades del Servidor Web Apache, debido a que fue desarrollado específicamente para trabajar con este Servidor Web y se instala como módulo externo. Este firewall proporciona una capa de seguridad adicional y brinda la capacidad de analizar el tráfico de la red mediante un filtro de solicitudes para detectar actividades sospechosas y prevenir el procesamiento de solicitudes HTTP maliciosas. ModSecurity proporciona protección contra diversos ataques que pueden afectar a servidores Web Apache.

Una vez habilitado el filtrado de solicitudes de ModSecurity, toda solicitud HTTP que llega al Servidor Web es capturada y analizada antes de ser procesada. El análisis es realizado con base en un conjunto de reglas; como consecuencia, si una solicitud no cumple con las reglas configuradas, es rechazada. Las peticiones son normalizadas antes de ser analizadas. La normalización consiste en modificar cuidadosamente la entrada, para hacer un control sobre el conjunto de símbolos utilizados, y de este modo evitar ataques producto de la manipulación del formato de la solicitud como son ataques de inyección de código o ataques de evasión.

Como ya se mencionó, ModSecurity necesita de un conjunto de reglas, las cuales pueden ser reglas básicas o avanzadas, sin embargo, cada caso específico debe ser analizado cuidadosamente para evitar la configuración de reglas innecesarias o inconvenientes, debido al impacto en el rendimiento del Servidor Web Apache.

A pesar de que este proyecto de investigación está orientado al estudio de la configuración del Servidor Web Apache, y no considera la seguridad en la red, el acceso físico, la seguridad en el *host*, ni la seguridad en las aplicaciones Web, se siguieron las recomendaciones de [15] y [21], en las cuales se sugiere aislar el Servidor Web Apache del sistema operativo, en particular con el fin de mitigar el impacto que podría sufrir una organización si el Servidor Web se ve comprometido por un atacante. Este procedimiento se conoce como *Jail* (jaula), y es apropiado para ambientes tipo Unix. En este proyecto la configuración de Apache fue llevada a un entorno aislado en forma exitosa, y se ha considerado como una buena práctica que debe ser incluida en toda configuración del Servidor Web Apache en producción.

Finalmente, y fuera del alcance de este proyecto, vale la pena mencionar una buena práctica, que consiste en elegir cuidadosamente la ubicación del Servidor Web, especialmente cuando se encuentra expuesto a redes públicas, como es el caso de Internet; es recomendable evitar la ubicación del servidor

en esta zona de acceso público. En su lugar, debe ser ubicado detrás de un *firewall* de frontera que pueda filtrar las solicitudes que van dirigidas al Servidor, exclusivamente hacia los servicios HTTP y HTTPS legítimos según los puertos establecidos en la configuración, usualmente los puertos 80 y 443, respectivamente. Lo anterior con el fin de evitar ataques sobre otros puertos abiertos o aplicaciones expuestas en el Servidor Web que no hayan sido considerados en el proceso de aseguramiento, siendo esto ajeno a la responsabilidad del administrador del Servidor Web.

V. Conclusiones, Aporte Y Trabajo Futuro

Hay suficiente información publicada con respecto a la seguridad en la configuración del Servidor Web Apache. Aplicar estas configuraciones en escenarios reales no es difícil, pero se necesita conocimiento en diferentes ámbitos, no siempre muy populares, como son: sistemas operativos como Windows y tipo Unix, manejo de particiones del disco duro, conceptos de gestión de procesos, permisos y propiedad de los archivos, gestión de usuarios, y vocabulario sobre seguridad de la información.

La instalación del Servidor Web Apache, junto con la configuración por defecto, son insuficientes para ser utilizadas en un servidor que propenda por la seguridad de la información.

El aseguramiento del Servicio Web debe estar alineado con los procesos de calidad de una institución. Por lo tanto, en lugar de realizar esfuerzos individuales, debe definirse un ciclo de vida del proceso de aseguramiento del Servidor Web Apache, el cual debe incluir desde políticas de seguridad de la información hasta las acciones preventivas, reactivas y correctivas correspondientes a una cultura de seguridad informática centrada en el Servidor Web para el beneficio de la organización.

Un aporte de este proyecto de investigación es la metodología utilizada la cual supone la realización de las fases descritas en la sección III de este informe, soportado en los resultados de la *XI Encuesta Nacional de Seguridad Informática* [1], en la cual se puede evidenciar la necesidad de disponer de metodologías concretas y ágiles que puedan ser utilizadas por las organizaciones para mejorar sus condiciones de seguridad de la información relacionadas con el Servidor Web Apache.

Este trabajo representa un primer esfuerzo realizado desde la Universidad del Quindío en el área de la seguridad informática, pero, queda mucho por hacer en esta materia. Futuros proyectos podrían ser los siguientes: Construcción de reglas a la medida para el *firewall* de aplicación ModSecurity; Seguridad en aplicaciones sobre el Servidor Web Apache; Fortalecimiento de la seguridad a nivel de *host* de Servidores Web Apache; Aseguramiento de aplicaciones y módulos externos que se conectan con el Servidor Web Apache; y Aseguramiento en el perímetro de red de un Servidor Web Apache.

VI. REFERENCIAS

[1] A. Almanza, "Seguridad informática en Colombia. Tendencias 2010-2011", *Revista Sistemas* No. 119, pp. 46-73, 2011.

[2] Apache, *Apache*, 2011. Available: http://httpd.apache.org/

[3] AT&T, AT&T, 2012. Available: http://att.com/

[4] T. Berners-Lee, RFC 1945. Hypertext Transfer Protocol -- HTTP/1.0, 1996.

[5] Certicámara, Certicámara, 2012. Available: http://www.certicamara.com/

[6] E. Cole, R. Krutz, and J. Conley, *Network Security Bible*. Indianapolis, Indiana: Wiley Publishing, Inc. 2005.

[7] R. Fielding, RFC 2616. Hypertext Transfer Protocol -- HTTP/1.1, 1999.

[8] Geeknet, Securing a Linux/Apache Webserver with common opensource tools, 2012. Available: http://reg.accelacomm.com/servlet/Frs.FrsGetContent?id=40115117

[9] D. Gourley, B. Totty, M. Sayer, A. Aggarwal, and S. Reddy. HTTP: The definitive guide. USA: O'Reilly, 2012.

[10] M. Hofmann and L. Beamont. *Content Networking. Architecture, Protocols, and Practice*, The Morgan Kaufmann Series in Networking.

[11] J. F. Kurose and K. W. Ross, *Computer networking: A top down approach*, 5th Edition ed. Addison-Wesley.

[12] MCI, MCI Communications, 2011. Available: http://www.mci.com/.

[13] Microsoft, Internet Explorer. Available: http://www.microsoft.com/spain/windows/internet-explorer/default.aspx

[14] Microsoft, Internet Information Server. Internet Information Server, 2011.

[15] T. Mobily, *Hardening Apache*. USA: Apress, 2004.

[16] ModSecurity. Available: http://www.modsecurity.org/

[17] M. D. Moreno Luzón, F. J. Peris, and T. González, *Gestión de la Calidad y Diseño de Organizaciones. Teoría y estudio de casos.* Pearson Educación. 2001.

[18] Mozilla, Firefox, 2011. Available : http://www.mozilla-europe.org/es/firefox/.

[19] Netcraft, 2011. Available: http://news.netcraft.com/

[20] Nikto, Nikto, 2011. Available: http://cirt.net/nikto2

[21] I. Ristic, *Apache Security*. Sebastopol, USA: O'Reilly, 2005.

[22] L. Sepúlveda, *Aproximación hacia un esquema de calidad a través de un control de versions para adecuada gestión de archivos a partir del uso de software libre,* Universidad Autónoma de Bucaramanga - Universidad Oberta de Catalunya. Bucaramanga: UNAB.

[23] W. Soyinka, *Linux Administration: A Beginner's* Guide, 5th ed. McGraw-Hill. 2008.

[24] A.Tanenbaum and D. Wetherall, *Computer Networks*, 5th ed. Prentice-Hall. 2010.

[25] Verisign, Verisign, 2012. Available: http://www.verisign.com/.