



**FACULTAD DE INGENIERÍA, ARQUITECTURA Y  
URBANISMO**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

**TESIS**

**EVALUACIÓN DE RENDIMIENTO CARGA Y  
STRESS DE SERVIDORES WEB LINUX, BAJO LA  
NORMA ISO/IEC 25000**

**PARA OPTAR EL TITULO PROFESIONAL DE  
INGENIERO DE SISTEMAS**

**Autor:**

**Bach. Culqui Briceño Willy Alberto**  
<https://orcid.org/0000-0002-7957-487X>

**Asesor:**

**Mg. Atalaya Urrutia Carlos William**  
<https://orcid.org/0000-0002-2761-4868>

**Línea de Investigación:**

**Infraestructura, Tecnología y Medio Ambiente.**

**Pimentel- Perú**  
**2022**

**EVALUACIÓN DE RENDIMIENTO CARGA Y STRESS DE SERVIDORES WEB  
LINUX, BAJO LA NORMA ISO/IEC 25000**

**APROBACIÓN DEL JURADO**

**Bach. Culqui Briceño Willy Alberto**

---

**Grado, Apellidos y nombres  
Autor**

**Mg. Atalaya Urrutia Carlos William**

---

**Grado, apellidos y nombres  
Asesor**

**Mg. Villegas Cubas Juan**

---

**Grado, apellidos y nombres  
Presidente de Jurado**

**Mg. Muro Nuñez Alejandro**

---

**Grado, apellidos y nombres  
Secretario de Jurado**

**Mg. Mejia Cabrera Ivan**

---

**Grado, apellidos y nombres  
Vocal de Jurado**

## DEDICATORIA

A la persona que siempre estuvo conmigo desinteresadamente y ahora  
está en el cielo mi papá.

**Enrique Cabrejos Barboza**

## **AGRADECIMIENTO**

A mis padres por todo su apoyo.

**Willy Culqui Briceño**

## **RESUMEN**

En la investigación se lleva a una evaluación de los servidores web bajo la plataforma Linux, con el fin de determinar la eficiencia bajo pruebas de rendimiento, carga y stress y siguiendo los lineamientos de evaluación de calidad dados por la norma internacional ISO/IEC 25000.

Los servidores web Linux cumplen funciones muy importantes en el flujo de la información y comunicación en todo el mundo, es así que resulta crítico conocer el grado de eficiencia de estos.

Para ello, se empezó por identificar cuáles son los servidores web Linux más utilizados, para que los resultados sean aplicables a la realidad. Luego se implementaron los servidores en la plataforma Linux y se aplicaron las pruebas pertinentes para su evaluación.

### **Palabras Clave**

Pruebas de performance, servidores web, plataformas Linux, rendimiento, eficiencia, pruebas de stress, pruebas de carga, norma ISO/IEC 25000.

## **ABSTRACT**

In this research an evaluation of the web servers under the Linux platform is carried out, in order to determine the efficiency under tests of performance, load and stress and following the guidelines of quality evaluation given by the international standard ISO / IEC 25000.

Linux web servers fulfill very important functions in the flow of information and communication throughout the world, so it is critical to know the degree of efficiency of these.

To do this, we began by identifying which Linux web servers are most used, so that the results are applicable to reality. Then the servers were implemented in the Linux platform and the relevant tests were applied for their evaluation.

### **Keywords**

Web servers, Linux platforms, performance, efficiency, performance tests, load tests, stress tests, ISO / IEC 25000 standard.

## ÍNDICE

APROBACIÓN DEL JURADO .....	ii
DEDICATORIA .....	iii
AGRADECIMIENTO .....	iv
RESUMEN .....	v
ABSTRACT .....	vi
ÍNDICE .....	vii
I: INTRODUCCIÓN.....	1
1.1. Realidad Problemática .....	1
1.2. Antecedentes de Estudio .....	2
1.3. Teorías relacionadas al tema .....	5
1.3.1. Servidor .....	5
1.3.2. Servidor Web.....	6
1.3.3. Protocolo HTTP .....	8
1.3.4. Servidor Web Apache.....	9
1.3.5. Servidor Web Nginx.....	9
1.3.6. Servidor Web Lighttpd .....	10
1.3.7. Pruebas para la evaluación de servidores web .....	10
1.3.8. Norma ISO/IEC 25000.....	12
1.4. Formulación del problema.....	13
1.5. Justificación e importancia de la Investigación.....	13
1.6. Hipótesis .....	15
1.7. Objetivos .....	15
1.7.1. Objetivo general.....	15
1.7.2. Objetivos específicos.....	16
II: MATERIAL Y METODO .....	16
2.1. Tipo y Diseño de la Investigación .....	16
2.1.1 Tipo de Investigación.....	16
2.1.2. Diseño de la Investigación.....	16
2.2. Población y Muestra .....	16
2.2.1. Población.....	16

2.2.2. Muestra.....	17
2.3. Variables y Operacionalización .....	17
2.4. Técnicas e instrumentos de recolección de datos .....	21
2.5. Procedimiento de análisis de Datos .....	22
2.6. Criterios Éticos .....	23
2.7. Criterios de Rigor Científico.....	24
III: RESULTADOS .....	25
3.1. Resultados en Tablas y Figuras .....	25
3.2. Discusión de los resultados .....	43
IV: CONCLUSIONES Y RECOMENDACIONES .....	51
4.1. Conclusiones.....	51
4.2. Recomendaciones.....	53
V: REFERENCIAS.....	54
ANEXOS .....	57



# I: INTRODUCCIÓN

## 1.1. Realidad Problemática

La información actualmente está alojada en internet. Según (Ackermann & Greenstein, 2018), información de todo tipo, se aloja en grandes repositorios denominados servidores web. Cuando alguien quiere acceder a esta información realiza una solicitud a los servidores web y estos responden entregando la información pedida, así como lo expone (Menasce & Almeida, 2001, págs. 268-269).

Existen problemas con el papel que juegan los servidores web dentro de todo el proceso del intercambio de información. Uno de ellos es el problema del consumo de recursos en el hardware donde están instalados, es decir, para cumplir con la función de entregar la información hacen uso de recursos de hardware como el consumo de CPU y de memoria, si el consumo es demasiado o no es óptimo entonces decantará en pérdidas para la empresa que proporciona el servicio e insatisfacción para la persona que realiza la petición de la información al no recibirla dentro del tiempo estimado (ya que se está forzando al hardware y esto provoca una baja en el rendimiento del mismo).

Se ha utilizado instrumentos según el estándar ISO/IEC 25000 (ISO, 2019) Factores de rendimiento y criterios Velocidad de respuesta, Consumo de recursos, Rendimiento efectivo total.

En donde se aplicó las herramientas siguientes, APACHE, NGINX, se instaló el software Apache Bench, HTOP para verificar el rendimiento en el so Ubuntu Server 16.04 LTS,

Según los procedimientos a considerar es el tiempo de respuesta de los servidores web frente a una petición realizada por los usuarios. Se puede creer que la petición de un usuario no representa mayor problema para el servidor, sin embargo, cuando la misma información es solicitada simultáneamente por cientos o miles de usuarios entonces los servidores web tienden a disminuir su eficiencia o en algunos casos colapsar, es decir, se ve afectada la escalabilidad del servidor web.

Otra cuestión a discutir es el rendimiento de los servidores web en función del tiempo, ya que mientras mayor sea el tiempo en el que el servidor web este resolviendo una petición, mayor será el tiempo de utilización del hardware (cpu, memoria, etc.), esto a veces termina afectando la eficiencia de los servidores web para realizar sus funciones.

Uno de los formatos que se ha utilizado es el Formato de recopilación de datos que especifica las características y especificaciones de los servidores web, formatos de recopilación de datos de las distribuciones Linux y una guía de observaciones según las normas ISO/IEC 25000.

La técnica para lograr optimizar se plantea un modelo de cola, que ordena las peticiones simultaneas de los usuarios desde la más antigua hasta la más reciente y se va ejecutar en ese orden.

Todos estos problemas que se dan en los servidores web, repercuten en bajas económicas y en insatisfacción y descontento para los usuarios debido al rendimiento ineficiente que en muchos casos se presenta.

## **1.2 Antecedentes de Estudio**

( Dabkiewicz, 2010) en la investigación *“Web Server Performance Analysis”* plantea la necesidad de establecer una clara diferenciación de performance entro los principales servidores web que existen. Primero según datos estadísticos de utilidad y crecimiento en los últimos años determina la evaluación para los servidores web Apache y Nginx.

Para estos servidores se determinó evaluar en cuanto al rendimiento con archivos estáticos, al rendimiento con archivos dinámicos y la medición del ancho de banda que necesitan o consumen, esto último se asocia directamente con la cantidad máxima de conexiones que el servidor puede manejar eficientemente sin comprometer sus funciones.

De los resultados se obtuvo que Nginx servidor web obtuvo una funcionabilidad optima utilizando ficheros estáticos, por otro lado, el server Apache uso ficheros dinámicos y se evidencio una mejor funcionabilidad. Con respecto al número de solicitudes que se pueden procesar, Nginx

podría manejar alrededor de 500 solicitudes por segundos más que Apache. Entonces, para un sitio con archivos mucho más grandes y con muchos visitantes, es necesario una conexión de red con mayor ancho de banda o un entorno de servidor múltiple donde la carga se distribuye entre diferentes servidores. La carga de la CPU estaba en ambos servidores alrededor del 30% durante una prueba con un archivo PHP.

(Xiao & Dohi, 2010) en su investigación establece la relación entre la tasa de error que es una de las medidas de fiabilidad representativas en los servidores web Apache y los parámetros del sistema para determinar el rendimiento del servidor web apache y desarrollar un modelo de probabilidad para describirlo.

Para esto se implementó un sistema simple cliente servidor utilizando como plataforma windows server en el cual se instaló y configuró el servidor apache en su versión 2.2.4, y un cliente conectado a una red LAN local. Que tenga que medir la tasa de error se accesa a una página seleccionada con la herramienta Jmeter para generar peticiones HTTP (carga para el servidor). Luego de realizadas las mediciones, se plantea un modelo de probabilidad lógica para determinar, explicar y minimizar la tasa de error.

Como resultados, se evidencia que la tasa de error aumenta con respecto al tiempo y disminuye considerablemente en el corto plazo.

(Elleithy & Komaralingam, 2000) en su investigación *“Using a Queuing Model to Analyze the Performance of Web Servers”* se plantearon el problema del rendimiento de los servidores web al tener peticiones simultáneas. Es decir, se verificó que cuando se hacían múltiples peticiones en un periodo de tiempo simultáneo, el servidor se volvía lento e ineficiente en su respuesta.

Para lograr optimizar el rendimiento de los servidores web ante estos casos, se planteó un modelo de cola, que ordenaba las peticiones simultáneas de los usuarios desde la más antigua hasta la más reciente y se iban ejecutando en ese orden. Para esto se analizó el tiempo de carga de un archivo en un servidor web, y se anotó que el tiempo de carga

aumente gradualmente hasta un punto, a partir de entonces, aumenta repentinamente hacia el infinito. Esto define que existe un límite superior en la capacidad de servicio de los servidores web, este límite depende directamente del tamaño de los archivos. Al determinar este límite se pudo realizar el modelo de cola.

Con el modelo de cola se redujo considerablemente el número de conexiones simultáneas y de esta manera se evitó el bloqueo de los servidores, situaciones que se producen cuando la carga del servidor se aproxima a la máxima capacidad.

(Castillo Fiallos, 2012) en su investigación determinó que no se conoce el rendimiento ni el potencial real de los diferentes servidores web. Así que se propuso el análisis y evaluación de Servidores Web virtualizados, ISS (Windows server 2008) y Apache, teniendo como métricas de evaluación la cantidad de recursos utilizados y las pruebas de rendimiento. Para esto se realizaron pruebas independientes de rendimiento de tráfico de redes con el comando PING ICMP, con carga y sin carga para 5 y 10 usuarios y usando las herramientas Colasoft y Web Server Stress Tools.

Los resultados de las pruebas indicaron que Apache obtuvo un 73.14% de eficiencia mientras que IIS obtuvo 53.78%, concluyendo que Apache es una excelente alternativa en la selección de un servidor web, ya que nos permite gestionar los recursos del servidor de una manera eficiente y así generar un mejor rendimiento en la concurrencia de los usuarios finales.

Tras la investigación, se recomienda el uso del servidor web apache para implementar Máquinas Virtuales sobre Windows server con especificaciones mínimas en cuanto a memoria 256Mb de RAM y espacio en disco 32 Mb.

(Minaya Cubillas & Mauricio, 2008) en su investigación *“Una Revisión de los Métodos de Pruebas para Aplicaciones Web”* investigan cuáles son los métodos más adecuados para realizar una evaluación y pruebas en aplicaciones web. Dentro de la clasificación de aplicaciones web también ubican a los productos software y los servicios web. Para la clasificación de

las pruebas se tomó como base la implementación de requerimientos de software, dentro de estos se enumeran un conjunto de pruebas a realizar.

De los requerimientos funcionales se obtiene las pruebas de funciones internas y externas que mayormente evalúan el software. Para las pruebas de requerimientos no funcionales se obtuvieron las pruebas de carga, rendimiento y stress que sirven para evaluar la eficiencia en cuanto al rendimiento de los servicios web.

La investigación realizada logra clasificar y establecer los tipos de pruebas a realizar para evaluar el software y los servicios web y obtener resultados objetivos de acuerdo al tipo de contexto que se desee.

(Kundu, 2012) en su investigación "*Web Testing: Tool, Challenges and Methods*" exponen el problema de que no existe una metodología que sirva de guía para realizar la adecuada evaluación a los productos web. Para esto primero se revisó y documentó la literatura donde se describen todos los métodos, técnicas y procedimientos.

A partir de esta revisión y estudio previo, se detallaron los métodos a seguir y las pruebas a realizar para obtener resultados ordenados, confiables y objetivos. Se obtuvieron diferentes tipos de pruebas para la evaluación: performance, funcional, usabilidad, seguridad y contabilidad. Dentro de cada una de esas pruebas se enumeraron indicadores o factores de medición que aseguran la evaluación objetiva de la eficiencia de los productos web.

### **1.3. Teorías relacionadas al tema**

#### **1.3.1. Servidor**

En palabras de (Sulyman, 2014), un servidor es la ejecución de un software capaz de atender las peticiones de un usuario al servidor y generar una petición de respuesta de acuerdo a la solicitud. Según (Marchionni, 2011, pág. 24), los servidores se pueden ejecutar en cualquier computadora,

incluso en computadoras dedicadas a las cuales se les conoce individualmente como «el servidor». En la mayoría de los casos una misma computadora puede realizar múltiples servicios. La ventaja de tener servidores dedicados es la seguridad. (Ayaz & Atul, 2015)

La arquitectura cliente-servidor es la modalidad mediante la cual opera un servidor. Comúnmente proveen servicios de mucha importancia dentro de una red, según (Marchionni, 2011, pág. 23) estos servicios pueden ser brindados a usuarios públicos y privados. Entre los tipos de servidores dedicados más utilizados encontramos base de datos, impresión, juegos, aplicaciones, archivo, correo y web.

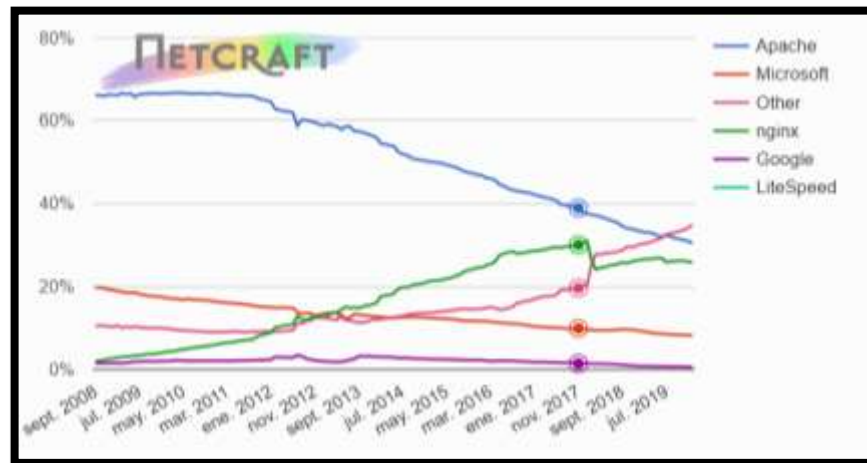
Los servidores web, correo y bases de datos son los que son usados por la mayoría de usuarios, tal como lo explica (Lowe, 2013, págs. 72-74) donde indica que la mayoría de servidores son usados para correo y archivos; pero que estos pueden realizar sus funciones en un mismo ordenador.

### **1.3.2. Servidor Web**

Es aquel que procesa una aplicación del lado del servidor, así lo sustentan (Tasneem & Ammar, 2012), y agregan que, un servidor web es capaz de realizar muchas conexiones bidireccionales o unidireccionales y síncronas o asíncronas con el cliente, generando o cediendo una respuesta en cualquier lenguaje o Aplicación del lado del cliente. El código recibido por el cliente es renderizado por un navegador web. Para la transmisión de todos estos datos suele utilizarse algún protocolo. Generalmente se usa el protocolo HTTP para estas comunicaciones, perteneciente a la capa de aplicación del modelo OSI. (Clifton, 2015, pág. 21)

El proceso que usa los servidores web para ejecutarse está basado en la arquitectura cliente/servidor, el hardware que almacena los portales web deben tener software que permita la ejecución de las aplicaciones alojadas en este. Según (NETCRAFT, 2019), los servidores web más importantes incluyen Apache, Nginx de la empresa del mismo nombre y luego está IIS de la corporación Microsoft. Otros servidores Web utilizan NetWare, Google (GWS) y los dominós de IBM. La siguiente imagen muestra los

resultados de la investigación realizada por (NETCRAFT, 2019) sobre los



servidores web que utilizan los sitios más activos en internet:

Figura 1. Servidores que más se utilizan en internet web server: Market share of the top million busiest sites. (NETCRAFT, 2019)

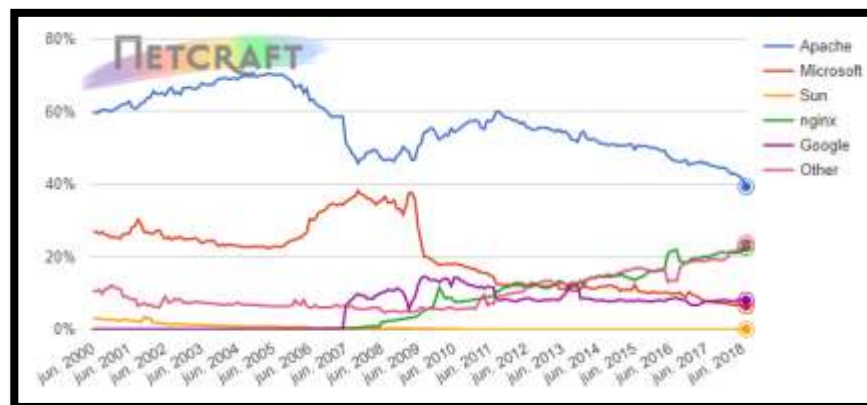


Figura 2. Servidores que más se utilizan en internet 2018. web server developers share of active sites. (NETCRAFT, 2019)

Para seleccionar un servidor Web se toman en cuenta ciertas consideraciones como: sistema operativo, capacidad de programación,

seguridad y algunas herramientas particulares (motor de búsqueda, publicación).

(Lowe, 2013, págs. 14-15), nos explica el despliegue de un protocolo HTTP; el cliente solicita una conexión TCP directa, luego al puerto HTTP envía un comando HTTP para solicitar un recurso y como respuesta se obtiene los datos solicitados pro el usuario final.

### **1.3.3. Protocolo HTTP**

Según (Ackermann & Greenstein, 2018), el Protocolo HTTP (Hypertext Transfer Protocol) realiza los intercambios de información entre los servidores y clientes web. La especificación completa del protocolo HTTP 1/0 está recogida en la RFC 1945, esta fue propuesta por Tim Berners-Lee para satisfacer las necesidades del sistema global de distribución de información (WWW - World Wide Web).

Estas peticiones están soportadas sobre los servicios de conexión TCP/IP, y funcionan de manera que un proceso ejecutado en servidor escucha un puerto de comunicación TCP (80), y espera las peticiones generadas para la conexión de los usuarios, tal como lo expone (Lowe, 2013, pág. 14) . Una vez que se establece la conexión, el protocolo TCP garantiza el intercambio y la comunicación de los datos.

En palabras de (Ludhin & Garza, 2017, pág. 42), la conexión entre cliente y servidor se basa en la transferencia de datos, donde el cliente solicita y el servidor responde enviando el código HTML, imágenes, etc.; esto basado en el protocolo HTTP (1.0).

HTTP contiene una variable llamada HTTPS (S = seguridad) que utiliza un protocolo de seguridad SSL para cifrar y autenticar el tráfico entre el cliente y servidor, esta es muy usada actualmente en los servidores web para generar la e-commerce. (Ludhin & Garza, 2017, pág. 15).



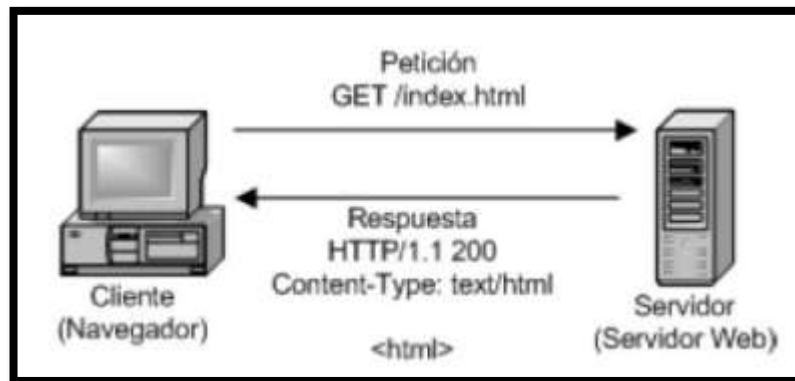


Figura 3. Funcionamiento del Protocolo HTTP. (Clifton, 2015)

#### 1.3.4. Servidor Web Apache

Es uno de los servidores más utilizados de Internet, con una estadística del 66%, según (Laurie & Laurie, 2009, pág. 18). Apache fue lanzado en abril de 1995, es un servidor que sigue evolucionando en la internet ya que este es open source. Sus objetivos son: diversidad en plataformas, celeridad, facilidad del desarrollo distribuido. Su código fuente está desarrollado en lenguaje de programación C con 185.000 sentencias de código fuente.

Según (Xialong, 2015, pág. 94), Apache es un servidor HTTP open source para Windows (Microsoft), Unix (BSD, GNU/Linux, etc) y otras, que desarrollan el protocolo HTTP/4.3. Apache permite la personalización (configuración) de los mensajes de errores, autenticación de base de datos y distribución de contenido; como deficiencia se toma la carencia de GUI que permita a los usuarios una ayuda en la configuración del servidor.

#### 1.3.5. Servidor Web Nginx

Nginx es un servidor ligero y con alto rendimiento que cumple además las funciones de un servidor proxy y de correo IMAP/POP3, así lo explica (Soni, 2016).

Además, el sitio oficial de Nginx (Nginx, 2019), explica que éste es considerado software libre y de código abierto. Por otro lado, estadísticas brindadas por (NETCRAFT, 2019), muestran la excelente gestión de los recursos de hardware de Nginx, de tal manera que su distribución en el

mundo informático aumento exponencialmente en la última década, tal es así que ha llegado a representar el 17.63% de utilización en el año 2017, solo por detrás de Apache.

Según (Nginx, 2019), Nginx es multiplataforma, y es adaptable a diferente tipo de hardware. Los sitios web y las empresas modernas usan en su mayoría Nginx como alternativa a apache, entre ellos tenemos: WordPress, Netflix, GitHub y partes de Facebook.

### **1.3.6. Servidor Web Lighttpd**

Lighttpd es un servidor web que creció en el mundo informático por ser poseedor de una flexibilidad muy alta, es decir, se adapta fácilmente a los entornos de desarrollo (Bogus, 2008, pág. 7). Está optimizado para ser rápido y para responder muy bien en entornos donde la velocidad de procesamiento y el tiempo de respuesta son muy importantes, así lo señala (Lind, 2014)

Al estar optimizado en cuanto al tiempo, su consumo de recursos de hardware es menor al de otros servidores (consumo de CPU y de memoria RAM), por lo tanto, en palabras de (Karayiannis, 2019), Lighttpd es apropiado para el hardware o entorno que presente ciertos problemas de carga.

Según la información brindada en la página oficial de Lighttpd (LightTTPD, 2017), este servidor web es de código abierto y se distribuye bajo la licencia BSD. Esta orientado para plataformas GNU/Linux y UNIX de forma oficial. Aunque también existen versiones para plataformas Windows.

### **1.3.7. Pruebas para la evaluación de servidores web**

Según (Glavich & Farrel, 2010, págs. 24-25), un servidor web tiene como servicio principal la entrega de información a partir de la solicitud o petición de un usuario. Cuando el servidor responde a la solicitud y entrega la información, hace uso de recursos del hardware que lo contiene, así consume un porcentaje de la CPU y un porcentaje de la memoria RAM, así lo explican (Jader, Zeebaree, & Zebari, 2019), estos factores son decisivos

si es que se quiere evaluar al servidor web. Es decir, en la medida en la que el consumo de recursos sea menor, mayor será el performance del servidor web.

En este orden de ideas, (Walkowiak, 2013) agrega que, el performance de un servidor web no puede quedar determinado solo por la medida en la que estos consumen recursos de hardware, sino que además existen para ello tipos de pruebas a realizar para tener una medida objetiva de la evaluación.

#### **a) Pruebas de Rendimiento**

Según el libro de buenas prácticas de rendimiento de HP (Hewlett-Packard., 2015, págs. 156-164), este tipo de pruebas son utilizadas para medir o evaluar características relacionadas al rendimiento, tales como consumo de CPU, consumo de memoria, estas pruebas se hacen generando o simulando cientos o miles de peticiones al servidor (cada petición representa a un usuario) en un determinado intervalo de tiempo.

#### **b) Pruebas de Carga**

Permite evaluar el tiempo necesario para llevar a cabo determinadas tareas y funciones bajo condiciones pre definidas, así como lo explican (Jugo, Kermek, & Mestrovic, 2014), es decir, las condiciones se fabrican generando una carga que será procesada por el servidor web en un tiempo determinado, a esto se conoce como tiempo de respuesta ante una carga específica que se le da al servidor.

#### **c) Pruebas de Stress**

En palabras de (Walkowiak, 2013), estas pruebas están hechas para medir como responde el servidor cerca o más allá de sus límites, es decir, se realizan conexiones simultáneas en el servidor para verificar como responde ante el estrés causado por los miles de peticiones.

### **1.3.8. Norma ISO/IEC 25000**

La ISO 25000 son un conjunto de normas internacionales que prestan un gran aporte para determinar el nivel de calidad de software de un determinado producto o servicio, como lo expresa (Valenciano, 2015, pág. 19).

En este orden de ideas, (ISO, 2019), explica que la finalidad del estándar ISO/IEC 25000 es administrar, enriquecer y sincronizar las series que cubren dos procesos principales: especificación de requerimientos y evaluación de QA del software, esta es monitoreada por el proceso de medición de QA software.

Las características y medición de la QA asociadas son utilizadas tanto para evaluar el producto de software como también para identificar los requerimientos mínimos de calidad. Según la información oficial brindada por (ISO, 2019), SQuaRE está formada por segmentada de la siguiente forma:

#### **a) ISO/IEC 2500n**

Formada por estándares que ayudan a definir modelos, métricas y referencias a las que se basa las otras divisiones de SquaRE.

#### **b) ISO/IEC 2501n.**

División del modelo de calidad. Según la documentación oficial (ISO/IEC 2510: 2011, 2011), el estándar que conforma esta división presenta un modelo de calidad detallado, incluyendo características para la calidad interna, externa y en uso.

#### **c) ISO/IEC 2502n.**

División de mediciones de calidad. Los estándares pertenecientes a esta división incluyen un modelo de referencia de calidad del producto software, definiciones matemáticas de las métricas de calidad y una guía práctica para su aplicación. Presenta aplicaciones de métricas para la calidad de software interna, externa y en uso.

**d) ISO/IEC 2503n.**

División de requisitos de calidad. Los estándares que forman parte de esta división ayudan a especificar los requisitos de calidad. Estos requisitos pueden ser usados en el proceso de especificación de requisitos de calidad para un producto software que va a ser desarrollado o como entrada para un proceso de evaluación. El proceso de definición de requisitos se guía por el establecido en la norma ISO/IEC 15288 (ISO, 2003).

**e) ISO/IEC 2504n.**

División de evaluación de la calidad. Estos estándares proporcionan requisitos, recomendaciones y guías para la evaluación de un producto software, tanto si la llevan a cabo evaluadores, como clientes o desarrolladores.

**f) ISO/IEC 25050–25099.**

Estándares de extensión SQuaRE. Incluyen requisitos para la calidad de productos de software “Off-The-Self” y para el formato común de la industria (CIF) para informes de usabilidad.

**1.4. Formulación del problema**

¿Cuál de los servidores web es el más eficiente en cuanto a rendimiento, carga y stress funcionando en plataformas Linux?

**1.5. Justificación e importancia de la Investigación**

Si el servidor web es eficiente realizara la entrega de información en tiempo óptimo, sin ningún error, y los niveles de consumo del hardware estarán aceptables, que repercute en el flujo dinámico de la información.

Realizar el análisis del rendimiento de servidores web Linux resulta de mucha utilidad debido al elevado porcentaje de utilización por los usuarios del mundo.

Es decir, Apache es el más usado y Nginx está escalando rápidamente para ubicarse detrás de Apache en cuanto a utilización. Es así según las

estadísticas resultantes de la investigación realizada en diciembre por Analistas de sitios web (NETCRAFT, 2019) se tiene que:

El servidor web apache ejecuta el 39.30% de los sitios web activos en Internet, lo que representa un total de más de 72 millones páginas y aplicaciones web. Mientras tanto, detrás de un 21.64% de los sitios web activos a nivel mundial se encuentra el servidor NGINX, con más de 41 millones de sitios activos.

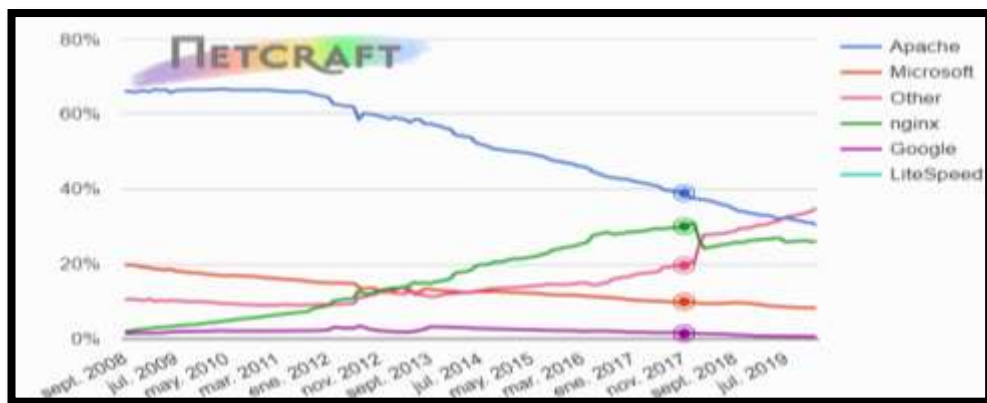


Figura 4 . web server developers:market share of the top million busiest sites. (NETCRAFT, 2019)

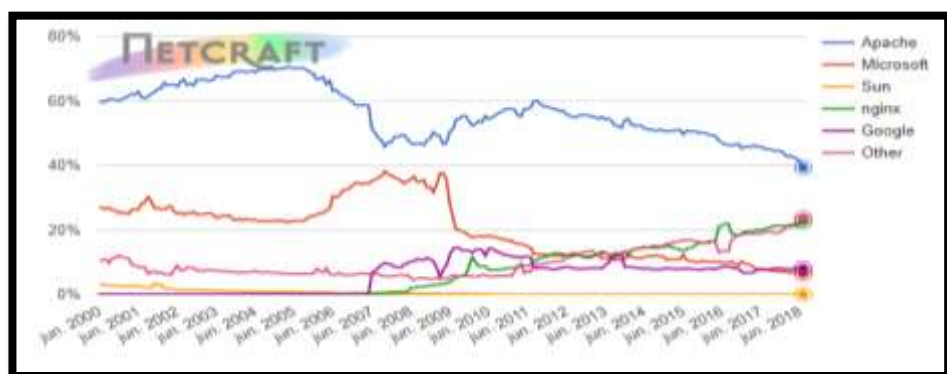


Figura 5. web server developers:Market share of active sites. (NETCRAFT, 2019)

Otro dato estadístico interesante que se toma en cuenta en este estudio es la participación de mercado, pero esta vez, enfocada únicamente en los sitios web con más actividad (la muestra es del primer millón de sitios más activos). Es en esta encuesta donde se puede ver el repunte de NGINX: para agosto de 2011, este servidor web representaba solo al 7% de los sitios más activos en la web; mientras que para julio de 2018 alcanza el 25%.

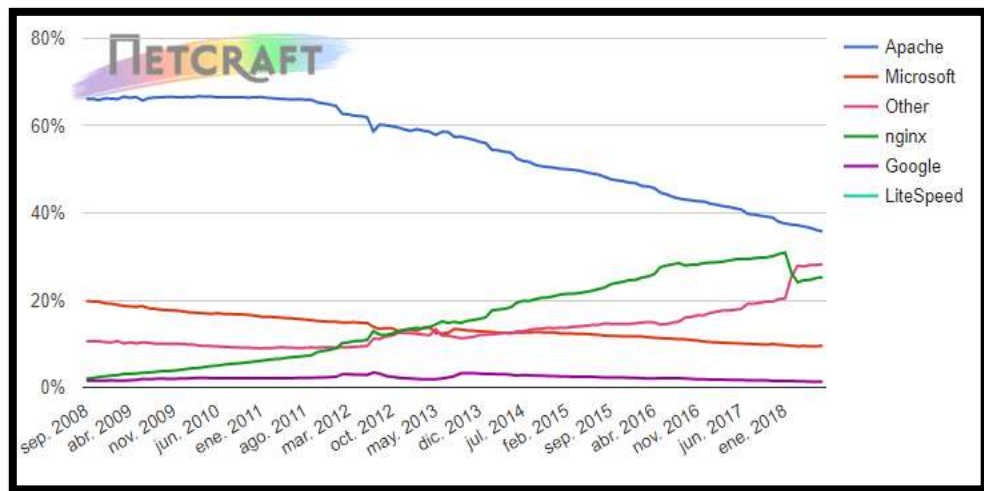


Figura 6. Web server developers:Market share of the top million busiest sites. (NETCRAFT, 2019)

## 1.6. Hipótesis

La evaluación del rendimiento, carga y stress de los servidores web Linux bajo la metodología ISO/IEC 25000 determinará cuál de los servidores web presenta una mayor eficiencia.

## 1.7. Objetivos

### 1.7.1. Objetivo general

Evaluar los servidores web Linux con la norma ISO/IEC 25000 que determine su grado de eficiencia.

### **1.7.2. Objetivos específicos**

- a. Determinar que servidores web Linux se van a tomar en cuenta para el estudio.
- b. Implementar y configurar los servidores web Linux sobre el servidor Ubuntu LTS 16.04.
- c. Elaboración de casos de prueba para la evaluación de rendimiento de los servidores, según la norma ISO/IEC 25000.
- d. Implementación y simulación de carga para el servidor (archivos), peticiones, conexión simultánea de usuarios. Para cumplir con los casos de prueba.
- e. Medir los parámetros establecidos, bajo la norma ISO/IEC 250000.
- f. Comparar los resultados obtenidos para todos los servidores web Linux.

## **II: MATERIAL Y METODO**

### **2.1. Tipo y Diseño de la Investigación**

#### **2.1.1 Tipo de Investigación**

El siguiente proyecto pertenece al tipo aplicado, porque para alcanzar los siguientes objetivos de la presente investigación los conocimientos reclutados son aplicados directamente.

#### **2.1.2. Diseño de la Investigación**

La presente investigación es tipo cuasi experimental, porque el investigador estableció una población y muestra.

### **2.2. Población y Muestra**

#### **2.2.1. Población**

Se tomó en cuenta los datos estadísticos de la investigación realizada en julio 2018 por Analistas de sitios web (NETCRAFT, 2019), los cuales



permiten conocer los principales servidores web tomando en cuenta los siguientes factores: participación en el mercado de los sitios web activos y participación en el mercado de los millones de sitios más buscados.

Los resultados arrojan los principales servidores web de acuerdo a los 3 factores señalados, estos resultados ya se explicaron y graficaron anteriormente en la presente investigación (entre las páginas 14 y 15).

Así la población queda determinada por los principales servidores web que pueden implementarse sobre una plataforma Linux: Apache, Nginx, Sun Java System Server, Google Web Server, Lighttpd.

### **2.2.2. Muestra**

Para establecerla se utiliza muestra por conveniencia del método no estadístico, así, se aplicó la investigación realizada en julio de 2018 por Analistas de sitios web (NETCRAFT, 2019), en la cual se toman en cuenta como factores de evaluación de servidores web: la participación en el mercado de los sitios web activos y la participación en el mercado de los millones de sitios más buscados, se puede ver en los resultados (explicados en las páginas 14 y 15 de la presente investigación) que Apache es el servidor más utilizado actualmente, pero Nginx se ubica detrás de Apache.

De esta manera, mediante el uso del método no estadístico de muestra por conveniencia, la muestra queda determinada por los servidores web Linux Apache y Nginx, por la relevancia de los servidores y para mayor utilidad de la investigación.

### **2.3. Variables y Operacionalización**

#### **a) Variable Independiente**

Evaluación de rendimiento, carga y stress de los servidores web Linux bajo la metodología ISO/IEC 25000

#### **b) Variable Dependiente**

Eficiencia de los servidores web

**c) Operacionalización:**

Tabla 1.

*Operacionalización*

<b>Variable independiente</b>	<b>Dimensiones</b>	<b>Indicadores</b>	<b>Descripción</b>	<b>Técnicas y métodos de recolección de datos</b>
Evaluación de rendimiento, carga y stress de los servidores web Linux bajo la metodología ISO/IEC 25000	Implementación de los servidores web linux	Porcentaje de utilización de los servidores web Linux	Cuota de mercado de servidores que se van a implementar para obtener lectura de parámetros	Análisis
	Aplicación de las pruebas de rendimiento, carga y	Capacidad de respuesta	Capacidad de respuesta a conexiones simultánea	Experimentación
				Observación

stress, bajo  
normas  
ISO/IEC  
25000

Utilización  
de  
Recursos

s de  
usuarios

Medida de  
la  
utilización  
de  
recursos  
de  
Hardware

Tabla 2.

*Variable Dependientes.*

<b>Variable dependiente</b>	<b>Dimensiones</b>	<b>Indicadores</b>	<b>Descripción</b>	<b>Técnicas y métodos de recolección de datos</b>
Eficiencia de los servidores web	Calidad en el servicio	Consumo de Cpu	Medida en la que el servidor consume CPU al procesar las peticiones	Análisis Experimentación Observación
		Consumo de memoria	Medida en la que el servidor consume memoria al procesar las peticiones	

Tiempo de  
respuesta

Tiempo que  
tarda el  
servidor en  
dar respuesta  
a las  
peticiones

## 2.4. Técnicas e instrumentos de recolección de datos

Se está utilizando en esta investigación los métodos de: observación, análisis y experimentación.

### a) Observación

Técnica visual que permite observar una situación real. Viene dado por el comportamiento de los servidores web, es decir, observar las cualidad y comportamientos en tiempo real de los servidores web una vez que ya han sido implementados en la plataforma Linux, las cuales se evidencian y documentan, además de las sugerencias que realizan los asesores y miembros del jurado calificador.

### b) Análisis

Consiste en extraer la información relacionada con la norma ISO/IEC 25000 y servidores web Linux de los diferentes documentos, libros, papers, revistas, etc.

### c) Experimentación

La investigación toma un problema de la realidad informática actual, enfocándose en la ejecución y desarrollo de los servidores web seleccionados en la plataforma Linux Ubuntu server, para luego medir la eficiencia de cada servidor bajo las métricas de la norma ISO/IEC 25000 utilizando las pruebas de rendimiento, carga y stress, de esta manera se validará la hipótesis de la presente investigación.

Por otra parte, los instrumentos usados para recolectar información en nuestra investigación, se realizan a continuación:

Se aplicaron formatos de recopilación de datos, obtenidos de formatos utilizados por otros investigadores de una síntesis de tablas y por información brindada por los sitios oficiales de los servidores web, que al ser software libre permiten tener a disposición mucha información. **(los formatos se pueden observar en los Anexos N°1 y N°2).**

Por lo tanto, se utilizarán guías de observación para documentar la información de los productos de la experimentación, estas guías han sido elaboradas en base a las especificaciones y características de los servidores web Linux dadas en sus sitios oficiales (**Anexo N°3**).

## **2.5. Procedimiento de análisis de Datos**

### **a) Análisis Documental**

El proceso realizado para recolectar información a través del análisis de la documentación en la presente investigación es la siguiente:

**a.1)** Recopilar y analizar las características y especificaciones de los servidores web, tales como: tipo de archivos, cantidad de peticiones que procesan por unidad de tiempo, tipo de archivos que resuelven, sistemas operativos en los que actúan, etc. Estos datos obtenidos serán de los sitios oficiales de los servidores y de las informaciones previas documentadas.

**a.2)** Recolectar y analizar los métodos y técnicas de análisis de servidores web utilizadas para determinar la eficiencia de los servidores seleccionados, las cuales han sido propuestas y desarrolladas en investigaciones realizadas anteriormente.

**a.3)** Disponer y realizar el análisis de los diferentes servidores web que se emplean que se pueden implementar en plataformas Linux, determinar la naturaleza de su funcionamiento y las características propias; esta información se obtendrá de los sitios oficiales de los servidores web y de las investigaciones anteriormente realizadas.

**a.4)** Recopilar y analizar las metodologías para evaluación de calidad y eficiencia de servidores web, los datos serán obtenidos de la revisión de las normas ISO internaciones de calidad del software.

### **b) Observación**

Observar el comportamiento y documentar cada uno de los servidores web luego de que han sido implementados sobre Linux, mediante los siguientes procesos:

**b.1)** Registrar el porcentaje de CPU consumido por cada uno de los servidores web evaluados.

**b.2)** Registrar el tiempo de respuesta a las peticiones para cada uno de los servidores web evaluados

**b.3)** Registrar el porcentaje de memoria consumida por cada uno de los servidores web evaluados.

## **2.6. Criterios Éticos**

En este proyecto para desarrollarse se ha considerado los criterios éticos:

### **a) Criterio de la Confidencialidad**

La recopilación de los datos personales productos de esta investigación, se debe obtener con total profesionalidad de una manera lícita que a las personas involucradas no se pueda causar ningún daño; lo expone La ley N° 29733: “Ley de protección de datos personales”, en su Título IV: “Obligaciones del titular y del encargado del banco de datos personales” menciona: recopilar los datos personales por medios ilícitos, fraudulentos es penado por nuestro estado peruano. De la misma manera, esta prohibido usar los datos personales con la finalidad que ha motivado su recopilación, salvo que se genere con anticipación un procedimiento de anonimizarían, disociación, con el objetivo de no identificar al titular mediante el uso de datos. El siguiente paso que se explica es que los datos han dejado de ser pertinentes o necesarios con el fin para cual hubiesen sido recopilados, estos serán retirados del banco de datos.

### **b) Criterio de Conformabilidad**

Todo lo que se haga en esta investigación como resultados y afirmaciones con respecto a los servidores web Linux deberán ser confirmados y fiables por los investigadores, así como lo comunica el Código Deontológico del Colegio de Ingenieros del Perú en el Capítulo

III “Faltas Contra la Ética Profesional y Sanciones”, del Sub Capítulo II “De la Relación con El Público” en su Artículo 105:

Los ingenieros en los informes y las declaraciones son objetivos y veraces las opiniones y temas de ingeniería se expresarán cuando solamente se verifica un adecuado análisis y conocimiento de los hechos, detalles científicos, técnicos suficientes y veraces.

## **2.7. Criterios de Rigor Científico**

### **a) Consistencia**

Los datos recolectados para este trabajo que son de carácter científico y formal. Se ha desarrollado un análisis utilizando datos con mucha profesionalidad aplicando las habilidades técnicas y conocimientos de ingeniería y de investigación para conservar la consistencia de los datos resuelve información útil y consistente.

### **b) Validez**

Los datos como resultado de los servidores web después de su funcionamiento serán adecuadamente analizadas y evaluados para obtener un resultado válido para comprobar la hipótesis.

### **c) Fiabilidad**

Las técnicas usadas e instrumentos de medición para adquirir los valores de los parámetros medibles de los servidores web, que muestran siempre los resultados con valores próximamente iguales en las mismas situaciones y circunstancias.

### **d) Objetividad**

A través de la experiencia con los servidores Linux, los investigadores obtendrán datos que permitirán analizarlos para obtener los resultados de la presente investigación.



### III: RESULTADOS

El presente capítulo permitirá organizar, evidenciar e interpretar los datos obtenidos durante las pruebas realizadas para determinar la eficiencia de los servidores web Linux: Apache y Nginx.

#### 3.1. Resultados en Tablas y Figuras

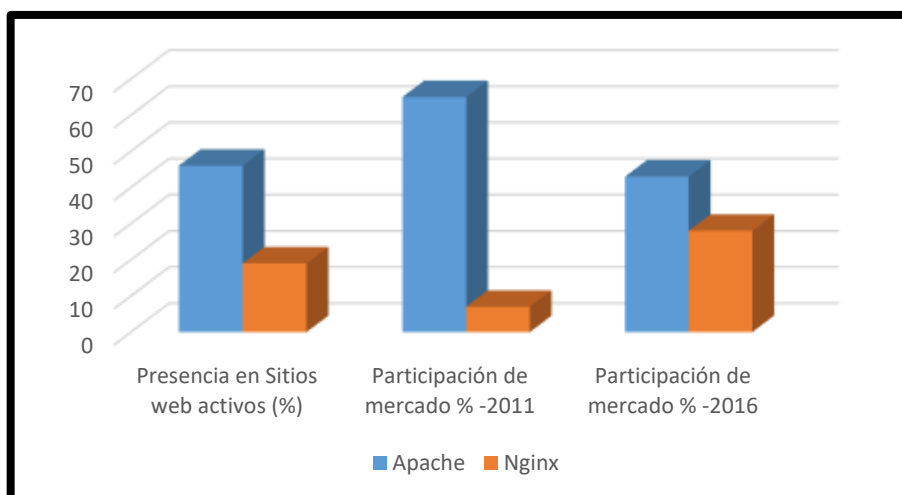
##### a) Porcentaje de utilización de los servidores web Linux

La tabla y gráfico siguientes, muestran los datos recopilados en la investigación, los cuales resumen la utilización e importancia actuales de los servidores web estudiados.

Tabla 3.

*Servidores web Linux.*

<b>SERVIDOR WEB</b>	<b>PRESENCIA EN SITIOS WEB ACTIVOS (%)</b>	<b>PARTICIPACIÓN DE MERCADO (En sitios web con más actividad) % - 2011</b>	<b>PARTICIPACIÓN DE MERCADO (En sitios web con más actividad) % - 2018</b>
APACHE	39	66	36
NGINX	23	7	25



*Figura 7. Porcentaje de Utilización de los servidores web Linux.*  
Elaboración Propia.

## b) Capacidad de respuesta

La capacidad de respuesta viene expresada por el tiempo en que tarda el servidor en responder a las peticiones. Tomando en consideración las condiciones de carga y stress establecidas en las pruebas, se expresan los resultados en cada uno de los escenarios establecidos.

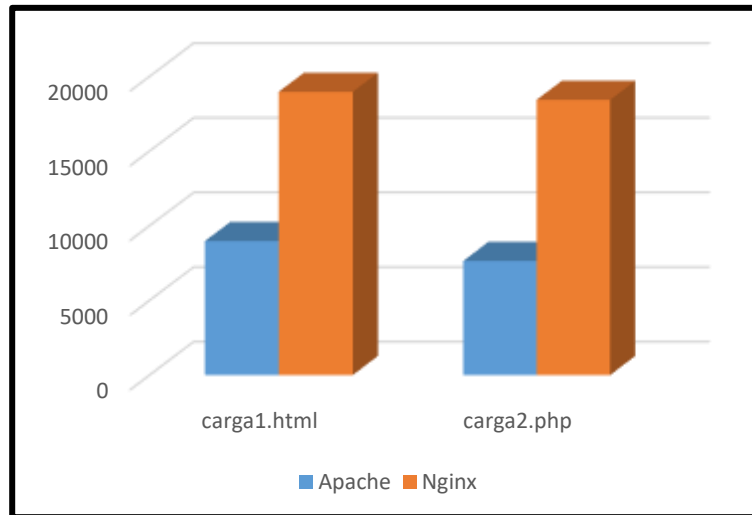
### ESCENARIO 1:

La tabla siguiente resume los resultados finales obtenidos para la evaluación de capacidad de respuesta de los servidores web evaluados:

Tabla 4.

*Escenario 1 Solicitudes / segundo – Apache vs Nginx.*

ARCHIVO	SERVIDOR WEB	N° DE SOLICITUDES /SEGUNDO	TIEMPO POR CADA SOLICITUD (ms)	TIEMPO TOTAL DE RESPUESTA (s)
Carga1.html	APACHE	8971.08	27.867	2.229
	NGINX	18974.67	13.175	1.054
Carga2.php	APACHE	7643.52	32.707	2.617
	NGINX	18431.23	13.564	1.085



*Figura 8. Tiempo por solicitud (ms) – Apache vs Nginx. Elaboración Propia.*

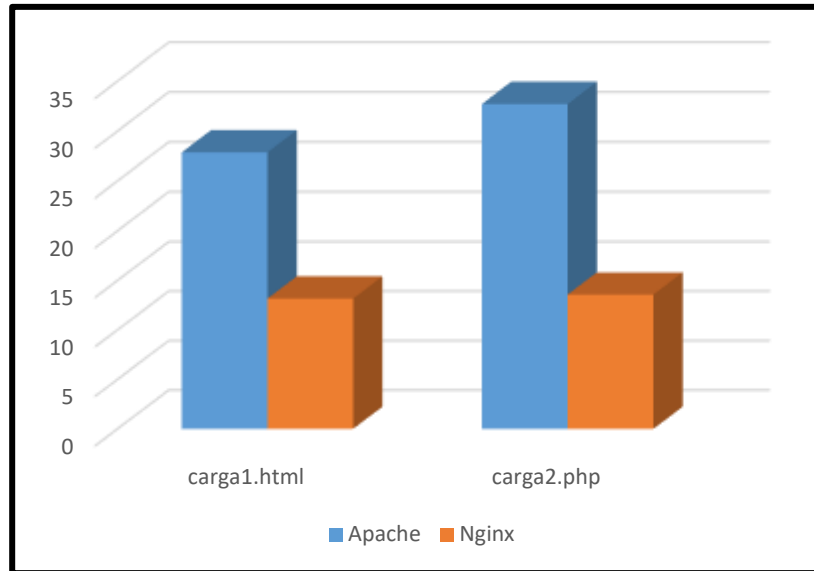


Figura 9. Tiempo Total de respuesta (s) - Apache vs Ngin. Elaboración Propia.

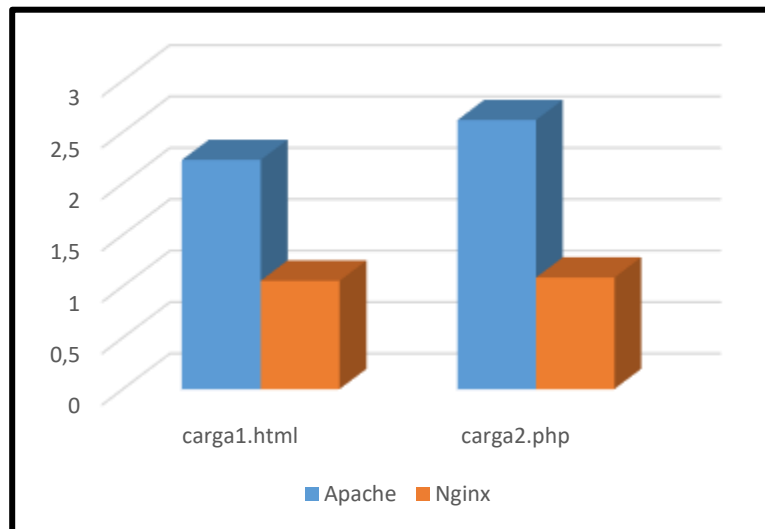


Figura 10. Apache vs Nginx. Elaboración Propia.

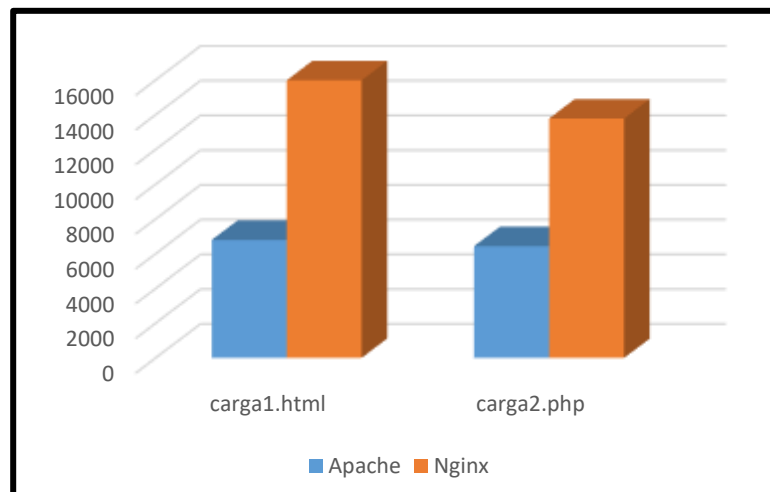
## ESCENARIO 2:

La tabla siguiente resume los resultados finales obtenidos para la evaluación de capacidad de respuesta de los servidores web evaluados:

Tabla 5.

*Escenario2.*

ARCHIVO	SERVIDOR WEB	N° DE SOLICITUDES /SEGUNDO	TIEMPO POR CADA SOLICITUD (ms)	TIEMPO TOTAL DE RESPUESTA (s)
Carga1.html	APACHE	6792.46	73.611	2.944
	NGINX	15987.47	31.274	1.251
Carga2.php	APACHE	6432.57	77.729	3.109
	NGINX	13779.02	36.287	1.451



*Figura 11.* Solicitudes / segundo – Apache vs Nginx.

Elaboración Propia.

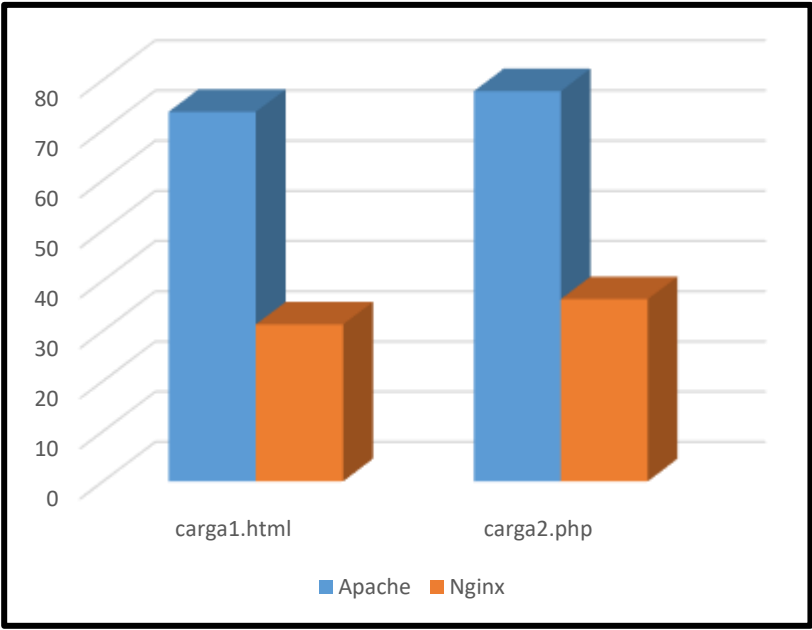


Figura 12. Tiempo por solicitud (ms) – Apache vs Nginx.  
Elaboración Propia.

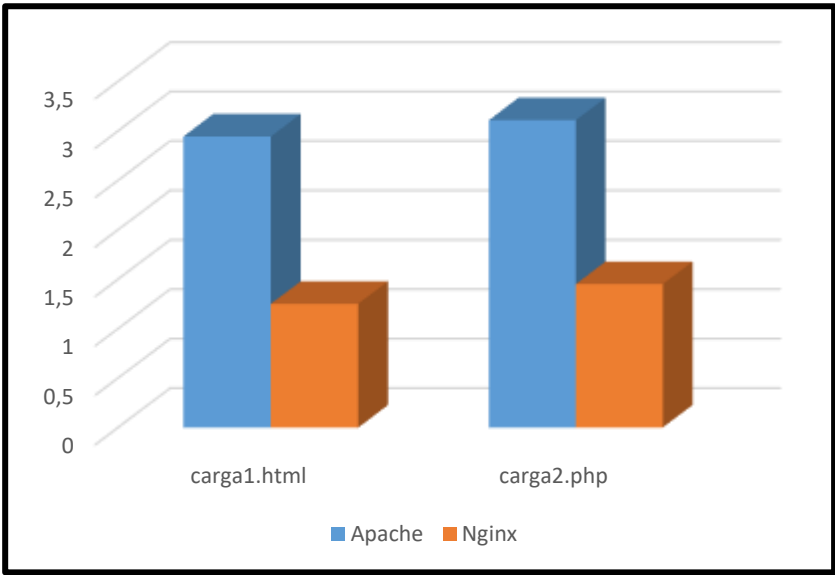


Figura 13. Tiempo Total de respuesta (s) - Apache vs Nginx.  
Elaboración Propia.

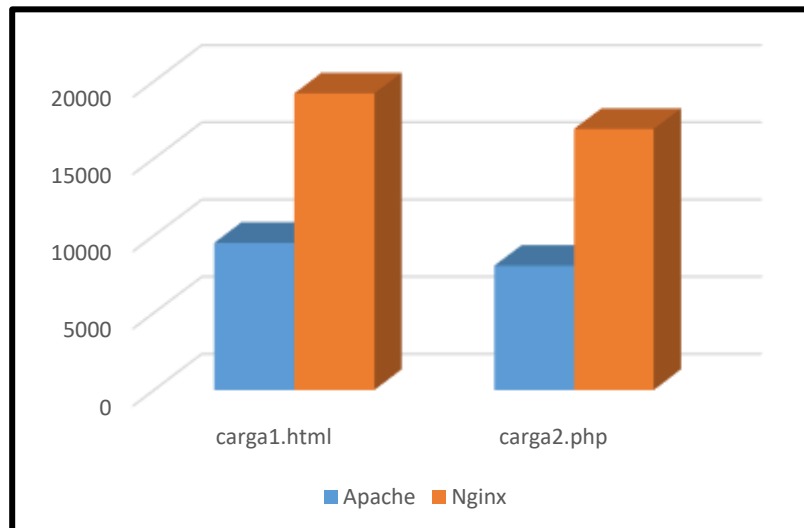
### ESCENARIO 3:

La tabla siguiente resume los resultados finales obtenidos para la evaluación de capacidad de respuesta de los servidores web evaluados:

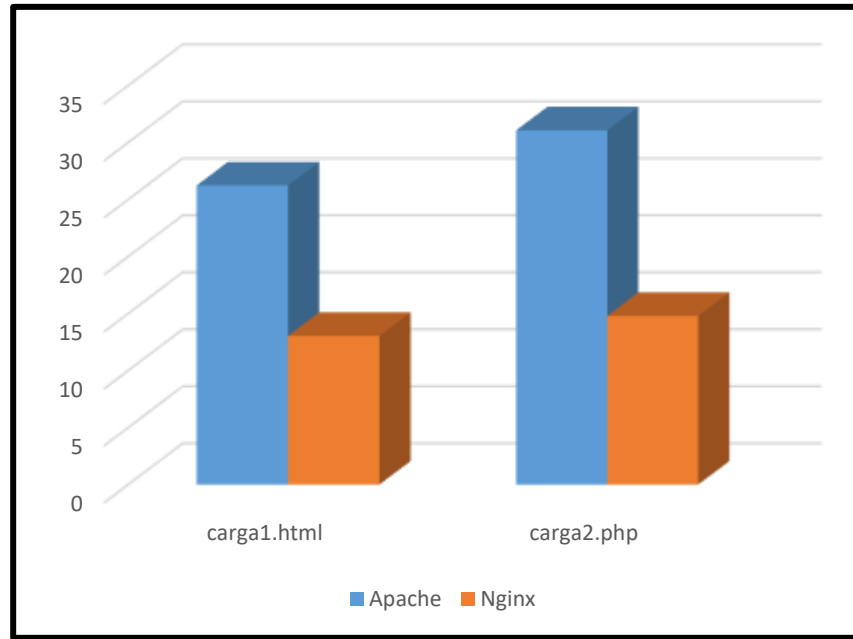
Tabla 6.

*Escenario 3.*

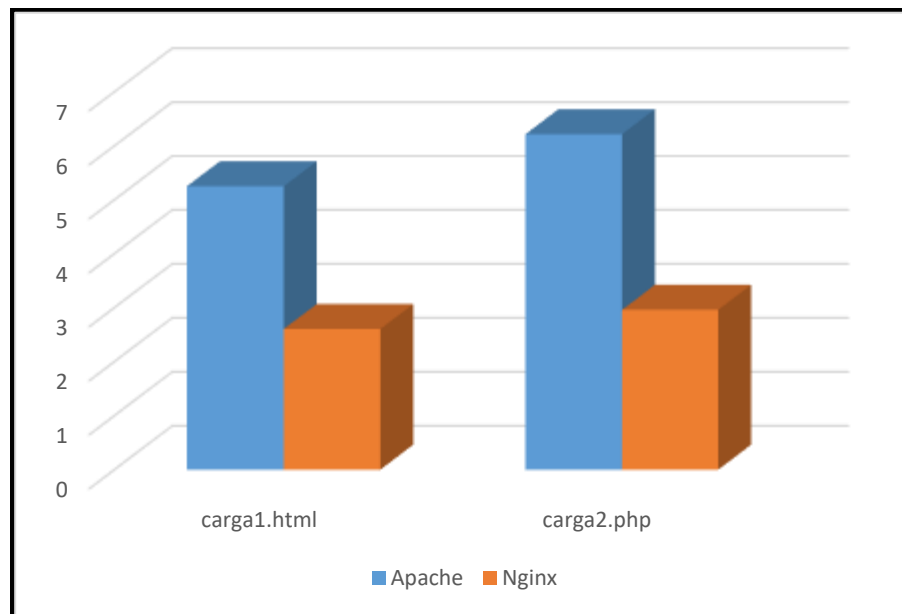
ARCHIVO	SERVIDOR WEB	N° DE SOLICITUDES /SEGUNDO	TIEMPO POR CADA SOLICITUD (ms)	TIEMPO TOTAL DE RESPUESTA (s)
Carga1.html	APACHE	9527.76	26.239	5.248
	NGINX	19209.41	26.239	5.248
Carga2.php	APACHE	8045	31.074	6.215
	NGINX	16899.55	14.793	2.959



*Figura 14.* Solicitudes / segundo – Apache vs Nginx.  
Elaboración Propia.



*Figura 15.* Tiempo por solicitud (ms) – Apache vs Nginx.  
Elaboración Propia.



*Figura 16.* Tiempo Total de respuesta (s) - Apache vs Nginx.  
Elaboración Propia.



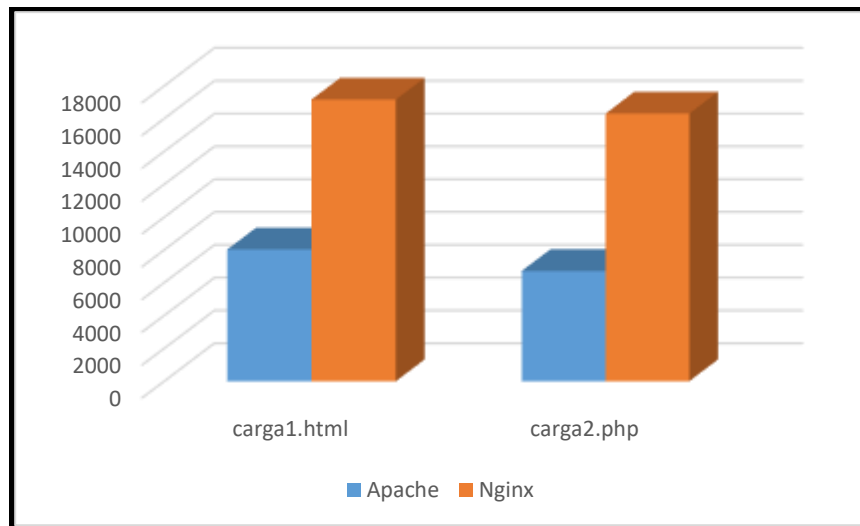
#### ESCENARIO 4:

La tabla siguiente resume los resultados finales obtenidos para la evaluación de capacidad de respuesta de los servidores web evaluados:

Tabla 7.

*Escenario 4.*

ARCHIVO	SERVIDOR WEB	N° DE SOLICITUDES /SEGUNDO	TIEMPO POR CADA SOLICITUD (ms)	TIEMPO TOTAL DE RESPUESTA (s)
Carga1.html	APACHE	8038.53	62.200	6.220
	NGINX	17168.60	29.123	2.912
Carga2.php	APACHE	6721.02	74.393	7.439
	NGINX	16315.77	30.645	3.065



*Figura 17.* Solicitudes / segundo – Apache vs Nginx.  
Elaboración Propia.

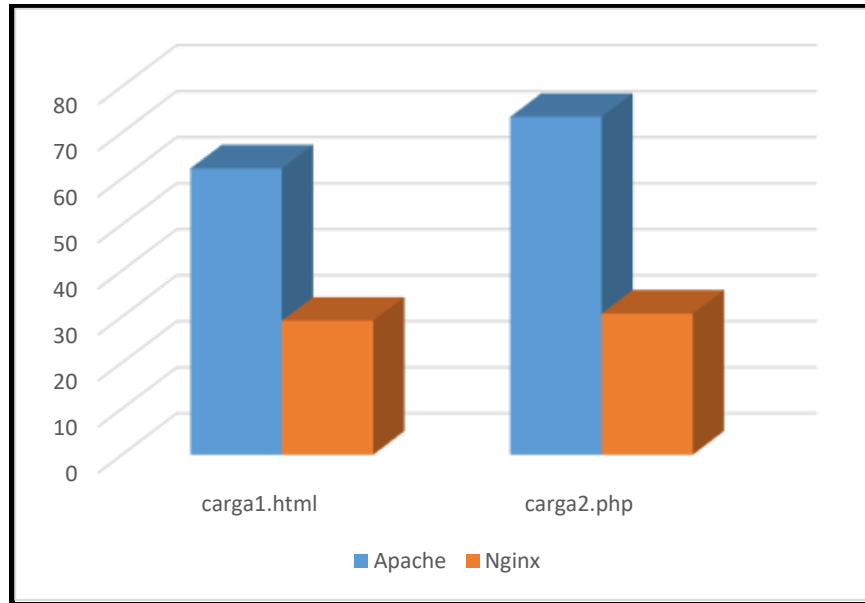


Figura 18. Tiempo por solicitud (ms) – Apache vs Nginx.  
Elaboración Propia.

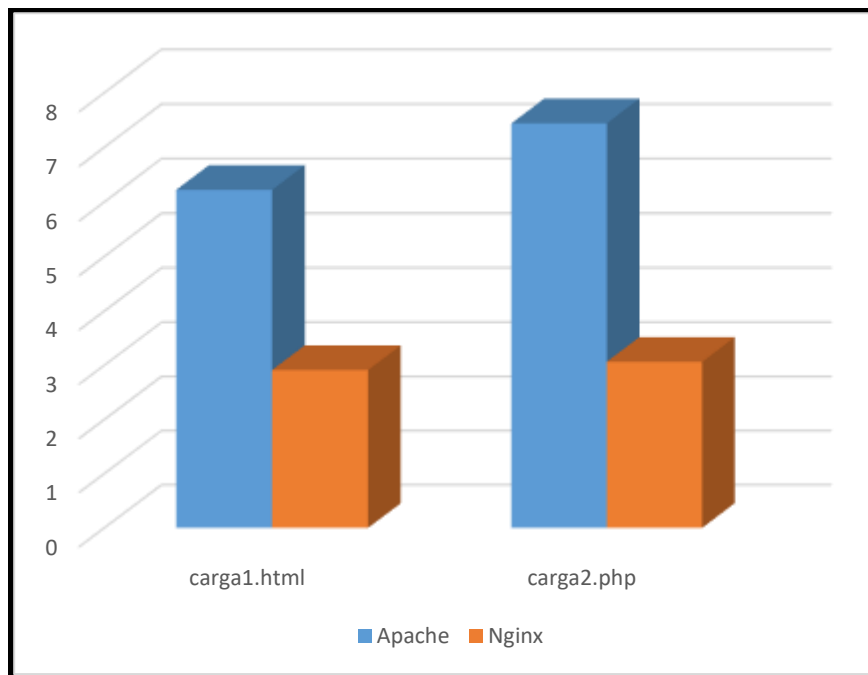


Figura 19. Tiempo Total de respuesta (s) - Apache vs Nginx.  
Elaboración Propia.

### ESCENARIO 5:

La tabla siguiente resume los resultados finales obtenidos para la evaluación de capacidad de respuesta de los servidores web evaluados:

Tabla 8.

*Escenario 5.*

ARCHIVO	SERVIDOR WEB	N° DE SOLICITUDES /SEGUNDO	TIEMPO POR CADA SOLICITUD (ms)	TIEMPO TOTAL DE RESPUESTA (s)
Carga1.html	APACHE	10055.82	24.861	9.944
	NGINX	18986.84	13.167	5.267
Carga2.php	APACHE	9439.13	26.486	10.594
	NGINX	18609.73	13.434	5.374

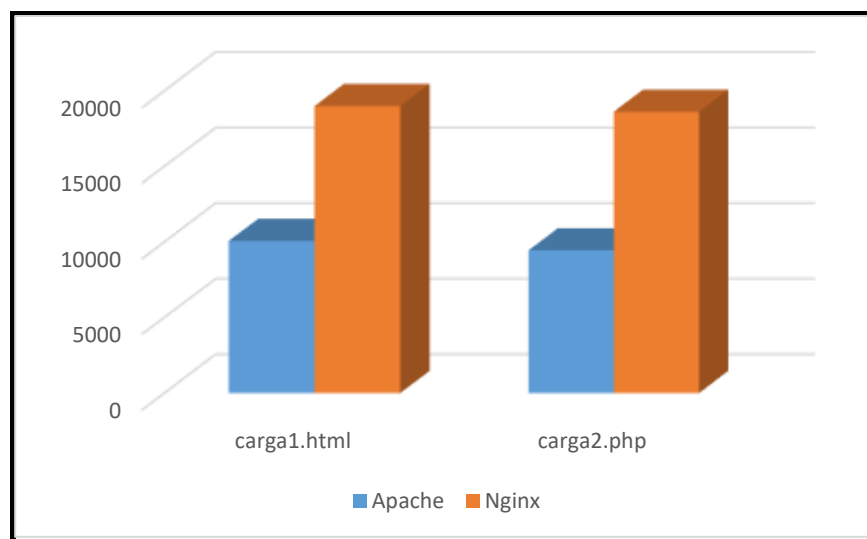
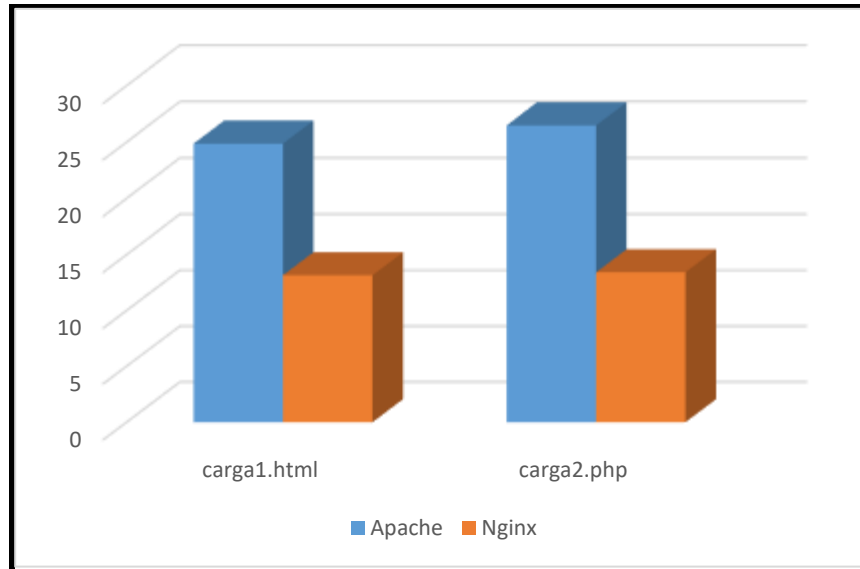
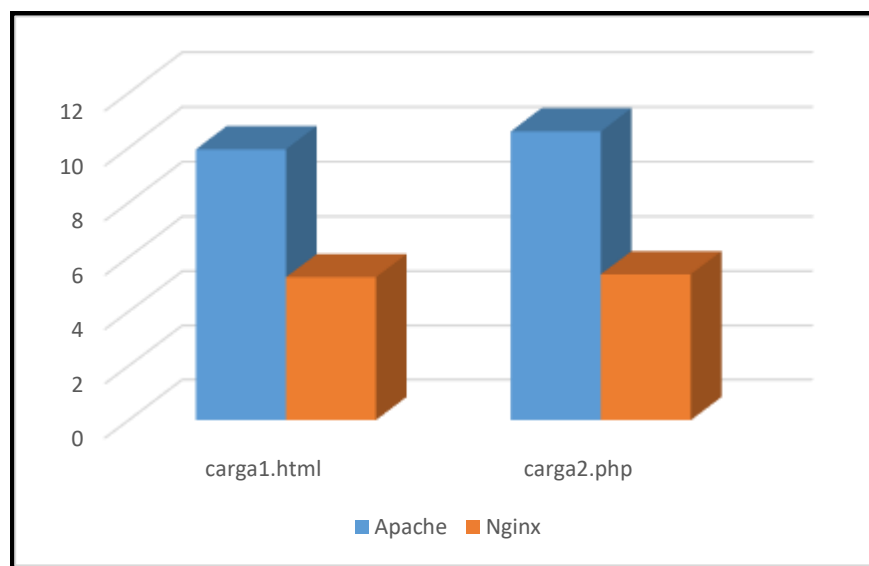


Figura 20. Solicitudes / segundo – Apache vs Nginx. Elaboración Propia.



*Figura 21.* Tiempo por solicitud (ms) – Apache vs Nginx.  
Elaboración Propia.



*Figura 22.* Tiempo Total de respuesta (s) - Apache vs Nginx.  
Elaboración Propia.

### ESCENARIO 6:

La tabla siguiente resume los resultados finales obtenidos para la evaluación de capacidad de respuesta de los servidores web evaluados:

Tabla 9.

ARCHIVO	SERVIDOR WEB	N° DE SOLICITUDES /SEGUNDO	TIEMPO POR CADA SOLICITUD (ms)	TIEMPO TOTAL DE RESPUESTA (s)
Carga1.html	APACHE	7222.13	69.232	13.846
	NGINX	16642	30.044	6.009
Carga2.php	APACHE	6724.83	74.351	14.870
	NGINX	16274.02	30.724	6.145

Escenario 6.

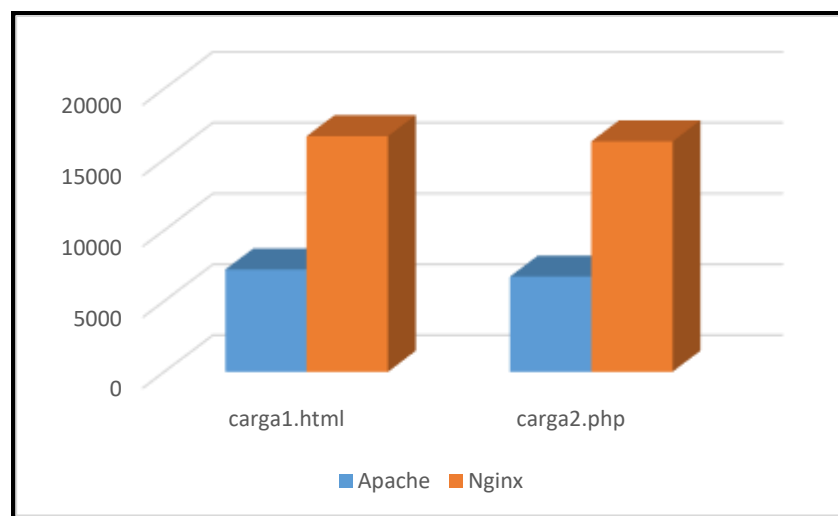


Figura 23. Solicitudes / segundo – Apache vs Nginx. Elaboración Propia.

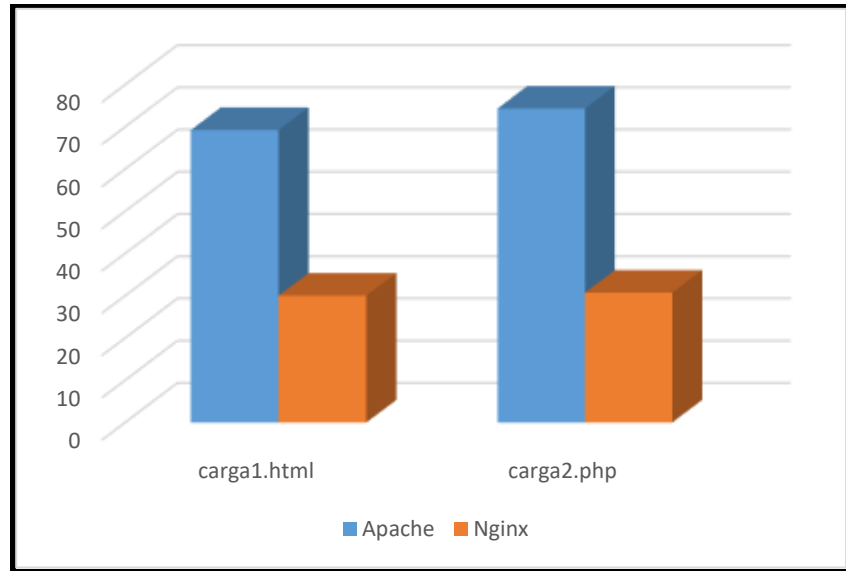


Figura 24. Tiempo por solicitud (ms) – Apache vs Nginx.  
Elaboración Propia.

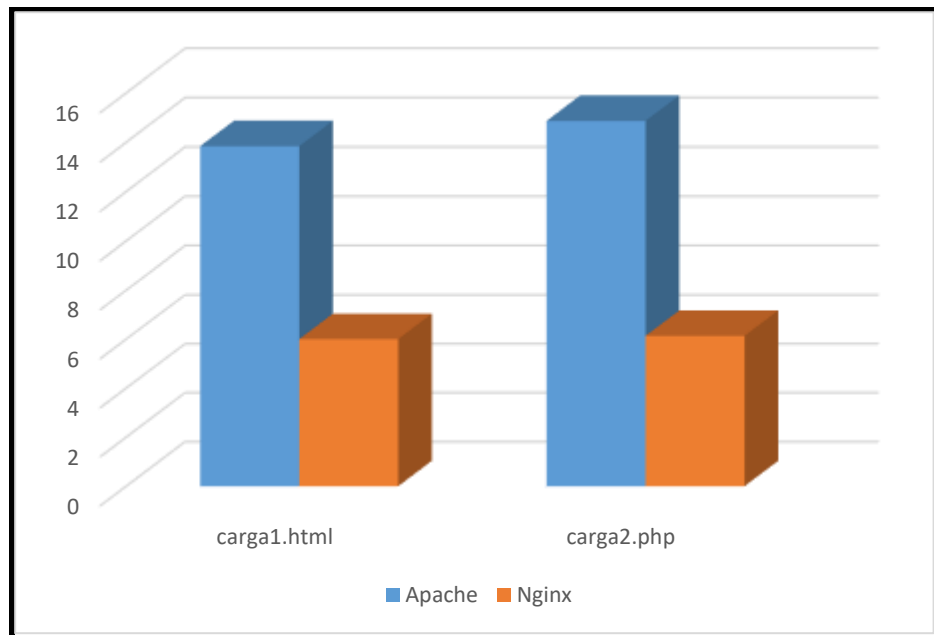


Figura 25. Tiempo Total de respuesta (s) - Apache vs Nginx.  
Elaboración Propia.

**c) Consumo de recursos**

Para el consumo de los recursos del hardware, se consideraron 2 indicadores

**Consumo de CPU – Carga1.html**

Los resultados obtenidos para este indicador se resumen a continuación

Tabla 10.

*Consumo de CPU - carga1.html.*

ARCHIVO	SERVIDOR WEB	CONSUMO DE CPU (1er núcleo) %	CONSUMO DE CPU (2do núcleo) %
Carga1.html	APACHE	100	100
	NGINX	86.7	48.3

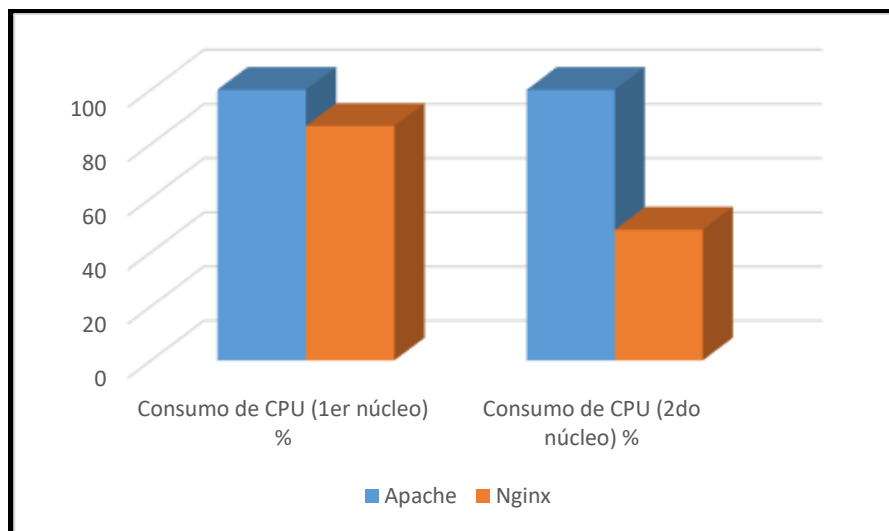


Figura 26. Consumo de CPU - carga1.html. Elaboración Propia.

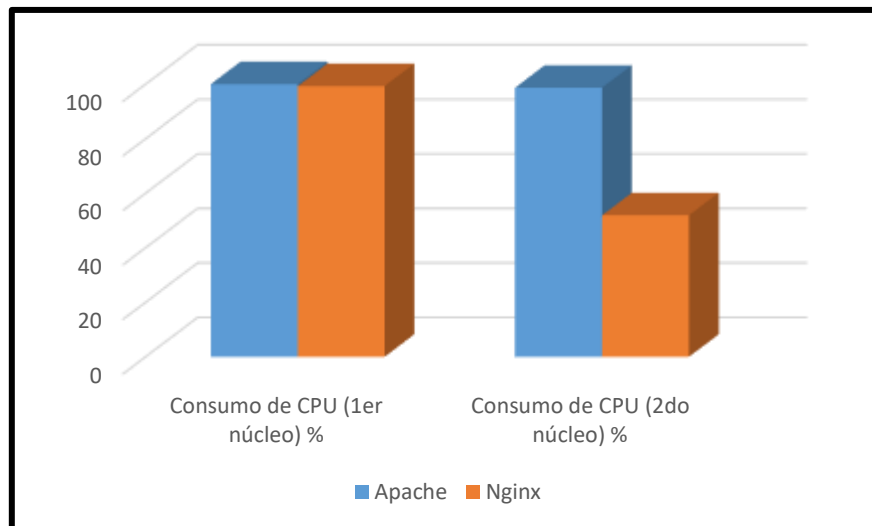
### Consumo de CPU – Carga2.php

Los resultados obtenidos para este indicador se resumen a continuación

Tabla 11.

*Consumo de CPU - carga2.php.*

ARCHIVO	SERVIDOR WEB	CONSUMO DE CPU (1er núcleo) %	CONSUMO DE CPU (2do núcleo) %
Carga2.php	APACHE	100	99.3
	NGINX	98.7	51.9



*Figura 27.* Consumo de CPU - carga2.php. Elaboración Propia.



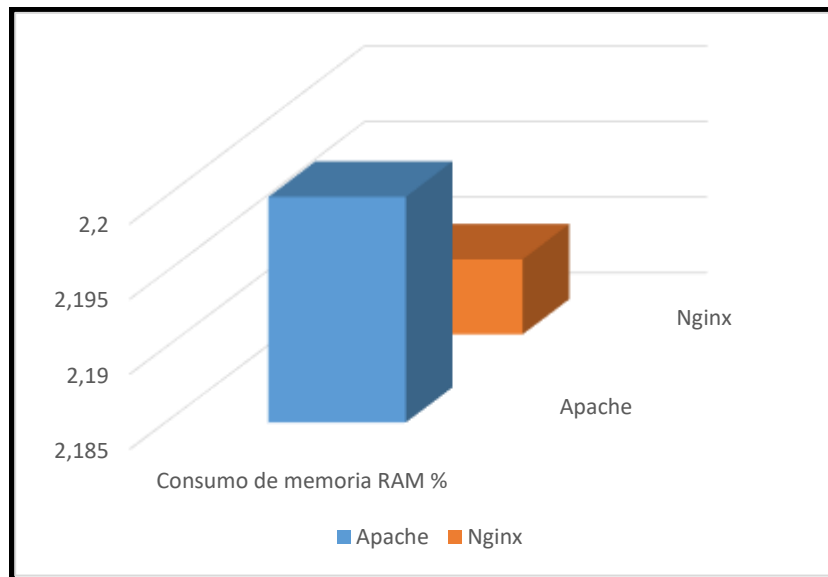
### Consumo de Memoria RAM

Igualmente, los resultados obtenidos para este indicador se resumen para cada escenario planteado anteriormente:

Tabla 12.

*Consumo de RAM - Carga1.html.*

SERVIDOR	CONSUMO DE MEMORIA RAM %
<b>WEB</b>	
APACHE	2.20
NGINX	2.19

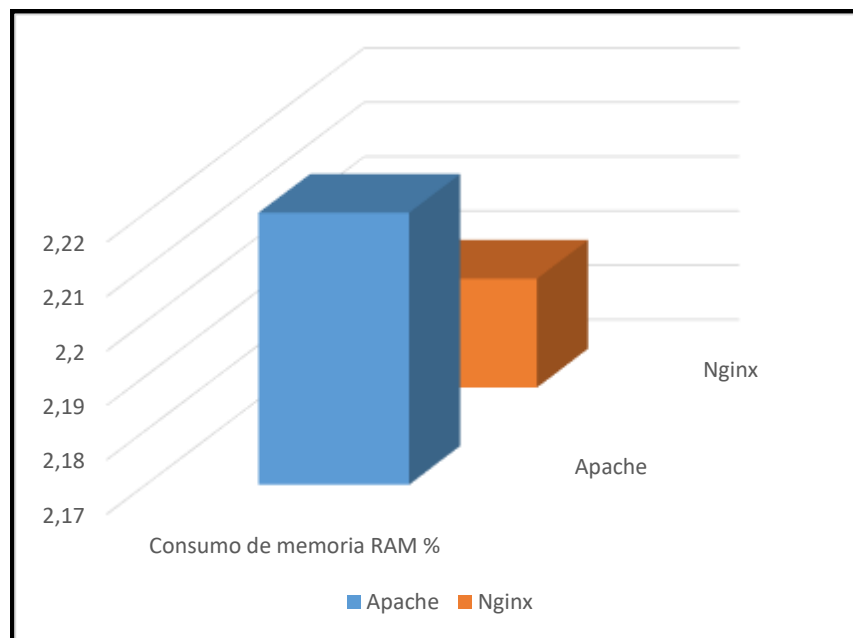


*Figura 28. Consumo de RAM - carga1.html. Elaboración Propia.*

Tabla 13.

*Consumo de RAM – Carga2.php.*

SERVIDOR	CONSUMO DE MEMORIA RAM %
WEB	
APACHE	2.22
NGINX	2.19



*Figura 29. Consumo de RAM – carga2.php. Elaboración Propia.*

### 3.2. Discusión de los resultados

Tabla 14.

Resultados Finales de las pruebas realizadas a los servidores web. Escenario  
N° 1 –

N° de peticiones: 20 000 - *N° de usuarios Concurrentes: 250.*

<b>ARCHIVO</b>	<b>SERVIDOR WEB</b>	<b>N° DE SOLICITUDES /SEGUNDO</b>	<b>TIEMPO POR CADA SOLICITUD (ms)</b>	<b>TIEMPO TOTAL DE RESPUESTA (s)</b>
Carga1.html	APACHE	8971.08	27.867	2.229
	NGINX	18974.67	13.175	1.054
Carga2.php	APACHE	7643.52	32.707	2.617
	NGINX	18431.23	13.564	1.085

Tabla 15.

Resultados Finales de las pruebas realizadas a los servidores web.

Escenario N° 2 - N° de peticiones: 20 000 - N° de usuarios Concurrentes:  
500.

<b>ARCHIVO</b>	<b>SERVIDOR WEB</b>	<b>N° DE SOLICITUDES /SEGUNDO</b>	<b>TIEMPO POR CADA SOLICITUD (ms)</b>	<b>TIEMPO TOTAL DE RESPUESTA (s)</b>
Carga1.html	APACHE	6792.46	73.611	2.944
	NGINX	15987.47	31.274	1.251
Carga2.php	APACHE	6432.57	77.729	3.109
	NGINX	13779.02	36.287	1.451

Tabla 16.

Resultados Finales de las pruebas realizadas a los servidores web.

Escenario N° 3 - N° de peticiones: 50 000 - N° de usuarios Concurrentes: 250.

<b>ARCHIVO</b>	<b>SERVIDOR WEB</b>	<b>N° DE SOLICITUDES /SEGUNDO</b>	<b>TIEMPO POR CADA SOLICITUD (ms)</b>	<b>TIEMPO TOTAL DE RESPUESTA (s)</b>
Carga1.html	APACHE	9527.76	26.239	5.248
	NGINX	19209.41	26.239	5.248
Carga2.php	APACHE	8045	31.074	6.215
	NGINX	16899.55	14.793	2.959

Tabla 17.

Resultados Finales de las pruebas realizadas a los servidores web.

Escenario N° 4 - N° de peticiones: 50 000 - N° de usuarios Concurrentes: 500.

<b>ARCHIVO</b>	<b>SERVIDOR WEB</b>	<b>N° DE SOLICITUDES /SEGUNDO</b>	<b>TIEMPO POR CADA SOLICITUD (ms)</b>	<b>TIEMPO TOTAL DE RESPUESTA (s)</b>
Carga1.html	APACHE	8038.53	62.200	6.220
	NGINX	17168.60	29.123	2.912
Carga2.php	APACHE	6721.02	74.393	7.439
	NGINX	16315.77	30.645	3.065

Tabla 18.

Resultados Finales de las pruebas realizadas a los servidores web.

Escenario N° 5 - N° de peticiones: 100 000 - N° de usuarios Concurrentes: 250.

<b>ARCHIVO</b>	<b>SERVIDOR WEB</b>	<b>N° DE SOLICITUDES /SEGUNDO</b>	<b>TIEMPO POR CADA SOLICITUD (ms)</b>	<b>TIEMPO TOTAL DE RESPUESTA (s)</b>
Carga1.html	APACHE	10055.82	24.861	9.944
	NGINX	18986.84	13.167	5.267
Carga2.php	APACHE	9439.13	26.486	10.594
	NGINX	18609.73	13.434	5.374

Tabla 19.

Resultados Finales de las pruebas realizadas a los servidores web.

Escenario N° 6 - N° de peticiones: 100 000 - N° de usuarios Concurrentes: 500.

<b>ARCHIVO</b>	<b>SERVIDOR WEB</b>	<b>N° DE SOLICITUDES /SEGUNDO</b>	<b>TIEMPO POR CADA SOLICITUD (ms)</b>	<b>TIEMPO TOTAL DE RESPUESTA (s)</b>
Carga1.html	APACHE	7222.13	69.232	13.846
	NGINX	16642	30.044	6.009
Carga2.php	APACHE	6724.83	74.351	14.870
	NGINX	16274.02	30.724	6.145

Tabla 20.

Resultados Finales de las pruebas realizadas a los servidores web.  
 CONSUMO DE CPU - N° de peticiones: 50 000 - N° de usuarios  
 Concurrentes: 500.

ARCHIVO	SERVIDOR WEB	CONSUMO DE CPU (1er núcleo) %	CONSUMO DE CPU (2do núcleo) %
Carga1.html	APACHE	100	100
	NGINX	86.7	48.3
Carga2.php	APACHE	100	99.3
	NGINX	98.7	51.9

Tabla 21.

Resultados Finales de las pruebas realizadas a los servidores web. CONSUMO DE MEMORIA RAM - N° de peticiones: 50 000 - N° de usuarios Concurrentes: 500.

ARCHIVO	SERVIDOR WEB	CONSUMO DE MEMORIA RAM %
Carga1.html	APACHE	2.20
	NGINX	2.19
Carga2.php	APACHE	2.22
	NGINX	2.19

Con los valores de la tabla anterior se pueden hacer las siguientes afirmaciones:

**Respecto al Porcentaje de Utilización de los servidores web analizados, Apache y Nginx. Se tiene que:**

Que la presencia en sitios web activos es del 39% para el servidor Apache, frente a un 23% para el servidor Nginx, lo cual determina que actualmente el servidor Apache es más utilizado que el servidor Nginx.

Que la cuota de mercado de los sitios web con más actividad desde el 2011 hasta el 2018 para el servidor Apache a variado de 66% a 36%. Mientras que la cuota de mercado del servidor Nginx en ese mismo periodo a variado de 7% a 25%. Lo cual indica que, en los sitios web más activos la utilización de Apache viene en descenso (un 30% en el periodo indicado), frente a un aumento del 18% en la utilización del servidor Nginx.

### **El análisis de respuesta de los servidores apache y Nginx en su capacidad,**

Se tiene que:

**En el escenario N° 1**, el número de solicitudes respondidas por segundo, ejecutando el archivo carga1.html, es de 8971.08 para Apache y de 18974.67 para Nginx, mientras que ejecutando el archivo carga2.php es de 7643.52 para Apache y de 18431.23 para Nginx.

Por otro lado, ejecutando el archivo carga1.html se tiene que el tiempo empleado por cada solicitud (en milisegundos) es de 27.867ms para Apache y de 13.175ms para Nginx; mientras que ejecutando el archivo carga2.php este tiempo es de 32.707ms para el servidor Apache y de 13.564ms para Nginx.

Así mismo respecto al tiempo total de respuesta, se tiene que al ejecutar el archivo carga1.html, el tiempo total de respuesta es de 2.229s para el servidor web Apache y de 1.054s para el servidor Nginx, mientras que al ejecutar el archivo carga2.php el tiempo total de respuesta es de 2.617s para Apache y de 1.085s para el servidor web Nginx.

**En el escenario N° 2**, el número de solicitudes respondidas por segundo, ejecutando el archivo carga1.html, es de 6792.46 para Apache y de 15987.47 para Nginx, mientras que ejecutando el archivo carga2.php es de 6432.57 para Apache y de 13779.02 para Nginx.

Por otro lado, ejecutando el archivo carga1.html se tiene que el tiempo empleado por cada solicitud (en milisegundos) es de 73.611ms para Apache y de 31.274ms

para Nginx; mientras que ejecutando el archivo carga2.php este tiempo es de 77.729ms para el servidor Apache y de 36.287ms para Nginx.

Así mismo respecto al tiempo total de respuesta, se tiene que al ejecutar el archivo carga1.html, el tiempo total de respuesta es de 2.944s para el servidor web Apache y de 1.251s para el servidor Nginx, mientras que al ejecutar el archivo carga2.php el tiempo total de respuesta es de 3.109s para Apache y de 1.451s para el servidor web Nginx.

**En el escenario N° 3**, el número de solicitudes respondidas por segundo, ejecutando el archivo carga1.html, es de 9527.76 para Apache y de 19209.41 para Nginx, mientras que ejecutando el archivo carga2.php es de 8045.41 para Apache y de 16899.55 para Nginx.

Por otro lado, ejecutando el archivo carga1.html se tiene que el tiempo empleado por cada solicitud (en milisegundos) es de 26.239ms para Apache y de 13.014ms para Nginx; mientras que ejecutando el archivo carga2.php este tiempo es de 31.074ms para el servidor Apache y de 14.793ms para Nginx.

Así mismo respecto al tiempo total de respuesta, se tiene que al ejecutar el archivo carga1.html, el tiempo total de respuesta es de 5.248s para el servidor web Apache y de 2.603s para el servidor Nginx, mientras que al ejecutar el archivo carga2.php el tiempo total de respuesta es de 6.215s para Apache y de 2.959s para el servidor web Nginx.

**En el escenario N° 4**, el número de solicitudes respondidas por segundo, ejecutando el archivo carga1.html, es de 8038.53 para Apache y de 17168.60 para Nginx, mientras que ejecutando el archivo carga2.php es de 6721.02 para Apache y de 16315.77 para Nginx.

Por otro lado, ejecutando el archivo carga1.html se tiene que el tiempo empleado por cada solicitud (en milisegundos) es de 62.200 ms para Apache y de 29.123ms para Nginx; mientras que ejecutando el archivo carga2.php este tiempo es de 74.393ms para el servidor Apache y de 30.645ms para Nginx.

Así mismo respecto al tiempo total de respuesta, se tiene que al ejecutar el archivo carga1.html, el tiempo total de respuesta es de 6.220s para el servidor



web Apache y de 2.912s para el servidor Nginx, mientras que al ejecutar el archivo carga2.php el tiempo total de respuesta es de 7.439s para Apache y de 3.065s para el servidor web Nginx.

**En el escenario N° 5**, el número de solicitudes respondidas por segundo, ejecutando el archivo carga1.html, es de 10055.82 para Apache y de 18986.84 para Nginx, mientras que ejecutando el archivo carga2.php es de 9439.13 para Apache y de 18609.73 para Nginx.

Por otro lado, ejecutando el archivo carga1.html se tiene que el tiempo empleado por cada solicitud (en milisegundos) es de 24.861ms para Apache y de 13.167ms para Nginx; mientras que ejecutando el archivo carga2.php este tiempo es de 26.486ms para el servidor Apache y de 13.434ms para Nginx.

Así mismo respecto al tiempo total de respuesta, se tiene que al ejecutar el archivo carga1.html, el tiempo total de respuesta es de 9.944s para el servidor web Apache y de 5.267s para el servidor Nginx, mientras que al ejecutar el archivo carga2.php el tiempo total de respuesta es de 10.594s para Apache y de 5.374s para el servidor web Nginx.

**En el escenario N° 6**, el número de solicitudes respondidas por segundo, ejecutando el archivo carga1.html, es de 7222.13 para Apache y de 16642.28 para Nginx, mientras que ejecutando el archivo carga2.php es de 6724.83 para Apache y de 16274.02 para Nginx.

Por otro lado, ejecutando el archivo carga1.html se tiene que el tiempo empleado por cada solicitud (en milisegundos) es de 69.232ms para Apache y de 30.044ms para Nginx; mientras que ejecutando el archivo carga2.php este tiempo es de 74.351ms para el servidor Apache y de 30.724ms para Nginx.

Así mismo respecto al tiempo total de respuesta, se tiene que al ejecutar el archivo carga1.html, el tiempo total de respuesta es de 13.846s para el servidor web Apache y de 6.009s para el servidor Nginx, mientras que al ejecutar el archivo carga2.php el tiempo total de respuesta es de 14.870s para Apache y de 6.145s para el servidor web Nginx.

Lo cual indica que en todos los escenarios el servidor web Nginx resuelve un mayor número de solicitudes por unidad de tiempo frente al servidor apache que es menos eficiente en este aspecto.

De la misma manera, considerando todos los escenarios, el servidor web Nginx es considerablemente más eficiente que el servidor Apache en cuanto a tiempo de respuesta.

**Respecto al consumo de recursos de los servidores web analizados, Apache y Nginx.** Se tiene que:

El consumo de CPU para el archivo de tipo HTML (carga1.html) es del 100% en el núcleo 1 y del 100% en el núcleo 2 para el servidor Apache, y para el servidor Nginx el consumo de CPU es del 86.7% en el núcleo 1 y del 48.3%. Mientras que para el archivo PHP (carga2.php) el consumo de CPU es de 100% y 99.3% en el núcleo 1 y 2 respectivamente para Apache y de 98.7% en el núcleo 1 y 51.9% en el núcleo 2 para Nginx. Los datos demuestran que el consumo de CPU de Apache es considerablemente mayor al consumo de Nginx, es decir Nginx presenta una mayor eficiencia que apache respecto al consumo de CPU.

El consumo de memoria RAM es de 2.20% para Apache y de 2.19% para Nginx, para el archivo carga1.html. Mientras que para el archivo carga2.php el consumo es de 2.22% para Apache y de 2.19% para Nginx. Lo cual indica que Nginx Es ligeramente más eficiente que Apache respecto al consumo de memoria RAM.

## IV: CONCLUSIONES Y RECOMENDACIONES

### 4.1. Conclusiones

- a) Se logró determinar los servidores web Linux más importantes para el estudio, en base a información recopilada y analizada, así, se puede decir que los dos servidores web Linux más utilizados en el mercado son: Apache y Nginx, además, actualmente el servidor web Apache es más utilizado que el se servidor web Nginx (46% para Apache y 19% para Nginx). Sin embargo, en los últimos años, la cuota de mercado en los sitios web más activos ha variado favorablemente para Nginx, con un aumento de 7% en el 2011 al 28% en el 2016; frente al descenso de Apache de un 65% en el 2011 al 43% en el 2016. Es, decir Nginx está escalando rápidamente en el mercado a la vez que apache desciende.
- b) Se logró implementar y configurar los servidores web Linux. Para ello, primero se instaló y configuró Ubuntu Server a través del asistente, luego se realizó la instalación y configuración de Apache y Nginx a través del terminal de comandos de Ubuntu Server. Se puede concluir que La plataforma Linux permite instalar y configuración de los servidores web sin mayores complicaciones.
- c) Se elaboraron los casos de prueba para la evaluación del performance. Por ello se usa la norma ISO/IEC 25000 que otorga lineamientos para la evaluación del producto y desarrollo del software. Para los fines de la investigación se hizo referencia específicamente a la división ISO/IEC 25020, la cual se especializa en la medición del producto software (que viene representado por los servidores web). Los factores de evaluación de rendimiento que se establecieron en base a la norma fueron: Tiempo de respuesta, performance y utilización de recursos.
- d) Se implementaron archivos para simular la carga para los servidores (2 cargas: 1 archivos HTML y 1 archivo PHP). Los archivos fueron ubicados en los directorios visibles de los servidores web. Además, se elaboraron

comandos Linux (ejecutables) para realizar las peticiones a los servidores, así como las conexiones simultáneas de los usuarios.

- e) La medición se realizó en ambos servidores web. Se midió el tiempo de respuesta por cada petición, la velocidad de procesamiento, el tiempo total de respuesta, el consumo de memoria RAM y el consumo de CPU. Para las mediciones se emplearon las herramientas de testing ApacheBench (Para las pruebas de carga y stress) y HTOP (para las pruebas de rendimiento).
- f) Se logró establecer una comparativa de los servidores web Linux estudiados. La comparación se realizó bajo los parámetros establecidos por las normas ISO/IEC 25000, realizando pruebas de rendimiento, carga y stress. Se puede concluir que en todas las pruebas realizadas Nginx es considerablemente más eficiente que Apache trabajando en plataformas Linux.

## 4.2. Recomendaciones

Tras obtener los resultados de esta investigación, se recomienda que:

- a) La Universidad Señor de Sipán debería considerar incluir gradual y sostenidamente en el plan curricular correspondiente, el estudio de Nginx como servidor Web de aplicaciones debido a la eficiencia que ha demostrado poseer frente a Apache y por el aumento rápido de su cuota de mercado.
- b) Se recomienda a APESOL (Asociación Peruana de Software Libre), promover la investigación referida a servidores web Linux, y estar pendientes de la aparición de nuevas tecnologías que puedan hacer variar los resultados expuestos en esta investigación.

## V: REFERENCIAS

- Castillo Fiallos, J. N. (2012). Estudio Comparativo del Rendimiento de Servidores Web de Virtualización sobre la Plataforma Windows Server 2008. *Lia Project*, 40.
- Echevarría Perez, E., & Abella Paumier, A. (2014). Testing como Práctica para Evaluar la Eficiencia en Aplicaciones Web. *Revista Latinoamericana de Ingeniería de Software*, 52.
- Elleithy, K. M., & Komaralingam, A. (2000). Using a Queuing Model to Analyze the Performance of Web Servers. *Department of Computer Science and Engineering*, 19.
- Kunda, D., Chihana, S., & Sinyinda, M. (2017). Web Server Performance of Apache and Nginx: A Systematic. *Computer Engineering and Intelligent Systems*, 34.
- Kundu, S. (2012). Web Testing: Tool, Challenges and Methods. *IJCSI International Journal of Computer Science Issues*, 28.
- Minaya Cubillas, A., & Mauricio, D. (2008). Una Revisión de los Métodos de Pruebas para Aplicaciones Web. *Revista de Ingeniería de Sistemas e Informática*, 47.
- Xiao, X., & Dohi, T. (2010). Estimating the Error Rate in an Apache Web Server System. *International Journal of Software Engineering and Its Applications*, 50.
- Xu, X., Xu, T., & Yin, Y. (2014). Performance evaluation model of Web servers based on response time. *Conference Anthology, IEEE*, 56.
- Menasce, D. A., & Almeida, V. F. (2001). *Capacity planning for web services: metrics, models and methods* (5 ed.). New York: Pearson Education. Obtenido de <http://catalogue.pearsoned.co.uk/samplechapter/0130659037.pdf>
- ISO. (2019). ISO 2500. Calidad de Software y Datos. Obtenido de La familia de normas ISO/IEC 25000: <https://iso25000.com/index.php/normas-iso-25000>
- Ljubuncic, I. (2014). *Apache Web Server. Complete Guide*. Obtenido de <https://sites.duke.edu/workblog/files/2014/12/www.dedoimedo.com-apache-web-server-lm.pdf>
- Ackermann, K., & Greenstein, S. (2018). *The State of Open Source Server Software*. Department for Econometrics and Business Statistics, Monash University, 40. Obtenido de [https://www.hbs.edu/faculty/Publication%20Files/The%20State%20of%20Open%20Server%20Software\\_v5\\_784980e1-7a8f-4a02-a3b7-711cf8c27394.pdf](https://www.hbs.edu/faculty/Publication%20Files/The%20State%20of%20Open%20Server%20Software_v5_784980e1-7a8f-4a02-a3b7-711cf8c27394.pdf)

- Dabkiewicz, S. (2010). Web Server Performance Analysis. Lia Project, 25. Obtenido de [https://www.os3.nl/\\_media/2011-2012/courses/lia/sebastian\\_dabkiewicz\\_-\\_lia-report-sebastian-dabkiewicz.pdf](https://www.os3.nl/_media/2011-2012/courses/lia/sebastian_dabkiewicz_-_lia-report-sebastian-dabkiewicz.pdf)
- Ludhin, S., & Garza, J. (2017). Learning HTTP / 2: A practical Guide. New York: O'Reilly. Obtenido de <https://www.imel.ba/edukacija/learninghttp2.pdf>
- Laurie, B., & Laurie, P. (2009). Apache the definitive Guide. California: O'Reilly. Obtenido de <https://theswissbay.ch/pdf/Gentoomen%20Library/Networking/Apache/O%27Reilly%20-%20Apache%20The%20Definitive%20Guide%203rd%20Edition.pdf>
- Xialong, L. (2015). Software Engineering and Information Technology. Guangxi: Indiana State University. Obtenido de <https://books.google.com.pe/books?id=sF7FCwAAQBAJ&printsec=frontcover#v=onepage&q&f=false>
- Nginx. (2019). Nginx. Obtenido de What is NGINX?: <https://www.nginx.com/resources/glossary/nginx/>
- Bogus, A. (2008). Lighttpd. Installing, compiling and securing this lightning-fast web server. Birmingham: Packt Publishing. Obtenido de <http://netoearth.com/book/lighttpd.pdf>
- LighTTPD. (2017). LighTTPD. Obtenido de Welcome to LighTTPD: <https://redmine.lighttpd.net/projects/lighttpd/wiki>
- Glavich, P., & Farrel, C. (2010). .NET Performance Testing and Optimization. Simple Talk Publishing. Obtenido de <http://assets.redgate.com/community/books/dotnet-performance-testing-and-optimization.pdf>
- Valenciano, J. (2015). Auditoría y Mantenibilidad de Aplicaciones según la ISO/IEC 25000. Madrid: Universidad Complutense de Madrid. Obtenido de [https://eprints.ucm.es/37485/1/AUDITOR%20C3%8DA%20MANTENIBILIDAD%20APLICACIONES%20SEG%20C3%9AN%20LA%20ISO\\_IEC%2025000.pdf](https://eprints.ucm.es/37485/1/AUDITOR%20C3%8DA%20MANTENIBILIDAD%20APLICACIONES%20SEG%20C3%9AN%20LA%20ISO_IEC%2025000.pdf)
- ISO/IEC 2510: 2011. (2011). SQUARE - System and software quality models. Obtenido de ISO/IEC 2510: 2011: <https://www.iso.org/standard/35733.html>
- NETCRAFT. (2019). Encuesta de servidores Web. Londres. Obtenido de <https://news.netcraft.com/archives/2019/09/27/september-2019-web-server-survey.html>
- Clifton, M. (2015). Web Servers. Carolina del Norte: Syncfusion. Obtenido de <https://www.syncfusion.com/ebooks/webservers>
- Hewlett-Packard. (2015). HP Performance Engineering - Performance Monitoring Best Practices. Hewlett-Packard Development Company. Obtenido de

[https://softwaresupport.softwaregrp.com/doc/KM01757194?fileName=hp\\_man\\_Mon\\_BP\\_1250\\_pdf.pdf](https://softwaresupport.softwaregrp.com/doc/KM01757194?fileName=hp_man_Mon_BP_1250_pdf.pdf)

- Marchionni, E. (2011). Administración de servidores. Buenos Aires: Fox Andina. Obtenido de <https://clasesdeseguridadinformatica.files.wordpress.com/2014/03/administrador-de-servidores.pdf>
- Lowe, D. (2013). Networking for Dummies. New Jersey: John Wiley & Sons, Inc. Obtenido de <http://www.nortonaudio.com/Ficheiros/1118474082.Netwo.pdf>
- Ayaz, S., & Atul, G. (2015). Analysis of various techniques for improving Web performance. International Journal of Advance Research in Computer Science and Management Studies, 120.
- Jader, O., Zeebaree, S., & Zebari, R. (2019). A State Of Art Survey For Web Server Performance Measurement And Load Balancing Mechanisms. International Journal of Scientific & Technology Research, 70. Obtenido de <http://www.ijstr.org/final-print/dec2019/A-State-Of-Art-Survey-For-Web-Server-Performance-Measurement-And-Load-Balancing-Mechanisms.pdf>
- Jugo, I., Kermek, D., & Mestrovic, A. (2014). Analysis and Evaluation of Web Application Performance Enhancement Techniques. researchgate, 104. Obtenido de [https://www.researchgate.net/publication/286776386\\_Analysis\\_and\\_Evaluation\\_of\\_Web\\_Application\\_Performance\\_Enhancement\\_Techniques](https://www.researchgate.net/publication/286776386_Analysis_and_Evaluation_of_Web_Application_Performance_Enhancement_Techniques)
- Karayiannis, C. (2019). The Lighttpd Web Server. Researchgate, 75. Obtenido de [https://www.researchgate.net/publication/333375088\\_The\\_Lighttpd\\_Web\\_Server](https://www.researchgate.net/publication/333375088_The_Lighttpd_Web_Server)
- Lind, D. (2014). Performance evaluation of HTTP web servers in embedded systems. KTH Industrial Engineering and Management, 100. Obtenido de <https://www.diva-portal.org/smash/get/diva2:724235/FULLTEXT01.pdf>
- Soni, R. (2016). Introduction to Nginx Web Server. Researchgate, 62. Obtenido de [https://www.researchgate.net/publication/306525506\\_Introduction\\_to\\_Nginx\\_Web\\_Server](https://www.researchgate.net/publication/306525506_Introduction_to_Nginx_Web_Server)
- Sulyman, S. (2014). Client - Server Model. IOSR Journal of Computer Engineering, 86.
- Tasneem, S., & Ammar, R. (2012). Performance Study of a Distributed Web Server: An Analytical Approach. Journal of Software Engineering and Applications, 156. Obtenido de <https://pdfs.semanticscholar.org/cae6/ec62c8ef395389f988ad899222a462b53004.pdf>
- Walkowiak, T. (2013). Behavior of Web Servers in Stress Tests. Advances in Intelligent Systems and Computing, 58. Obtenido de [https://www.researchgate.net/publication/287079927\\_Behavior\\_of\\_Web\\_Servers\\_in\\_Stress\\_Tests](https://www.researchgate.net/publication/287079927_Behavior_of_Web_Servers_in_Stress_Tests)



## ANEXOS

### ANEXO N°1: RESOLUCIÓN DE APROBACIÓN

 **UNIVERSIDAD  
SEÑOR DE SIPÁN**

FACULTAD DE INGENIERÍA, ARQUITECTURA Y URBANISMO

**RESOLUCIÓN N° 0054-2020/FIAU-USB**  
Chiclayo, 07 de febrero de 2020

**VISTO:**

El acta de reunión N° 0402-2020, de fecha 04 de febrero de 2020, para la ejecución del Trabajo de **"EVALUACIÓN DE RENDIMIENTO DE CARGA Y ESTRÉS DE SERVIDORES WEB LINUX, BAJO LA NORMA ISO/IEC 35000"**, presentado por el(los) estudiante(s) **CULQUI BRICEÑO, WILLY ALBERTO**, de la Escuela Profesional de **INGENIERÍA DE SISTEMAS**, y;

**CONSIDERANDO:**

Que, de conformidad con la Ley Universitaria N° 30220 en su artículo 48° que a letra dice: "La investigación constituye una función esencial y obligatoria de la universidad, que la fomenta y realiza, respondiendo a través de la producción de conocimientos y desarrollo de tecnologías a las necesidades de la sociedad, con especial énfasis en la realidad nacional. Los docentes, estudiantes y graduados participan en la actividad investigadora en su propia institución o en redes de investigación nacional o internacional, creadas por las instituciones universitarias públicas o privadas";



Estado a lo expuesto, y en uso de las atribuciones conferidas y de conformidad con las normas y reglamentos vigentes,



**SE RESUELVE:**

**ARTÍCULO 1°:** APROBAR, el Título de Proyecto de tesis denominada **"EVALUACIÓN DE RENDIMIENTO DE CARGA Y ESTRÉS DE SERVIDORES WEB LINUX, BAJO LA NORMA ISO/IEC 35000"** perteneciente a la línea de Investigación de **INFRAESTRUCTURA, TECNOLOGÍA Y MEDIO AMBIENTE**, a cargo del(los) estudiante(s) **CULQUI BRICEÑO, WILLY ALBERTO**, de la Escuela Profesional de **INGENIERÍA DE SISTEMAS**.

**ARTÍCULO 2°:** ESTABLECER, que la inscripción del Título de Proyecto de tesis se realice a partir de emitida la presente resolución, y tendrá una vigencia de 02 años

**REGÍSTRESE, COMUNÍQUESE Y ARCHÍVESE**

  
 **Universidad Señor de Sipán**  
Escuela Profesional de Ingeniería  
Arquitectura y Urbanismo  
Institucional USB N° 1500 S.A.C.

  
 **Plaza Carlos Wilson Andino Urzúa**  
Calle General Acuña 21, Frente al  
de Chiclayo, Chiclayo - Chiclayo  
Institucional USB N° 1500 S.A.C. **Chiclayo, Perú**

**ADMISIÓN E INFORMES**  
074 482833 - 074 482832  
**CAMPUS USB**  
CALLE GENERAL ACUÑA 21, FRENTE AL  
DE CHICLAYO, CHICLAYO - CHICLAYO  
**Chiclayo, Perú**

[www.usp.edu.pe](http://www.usp.edu.pe)

## ANEXO 2

### FORMATO DE RECOPIACIÓN DE DATOS

**Datos a Recopilar:** Características y especificaciones de los servidores web

**Objetivo:** Recopilar información e identificar las Características y especificaciones de los servidores web.

Tabla 22.

	<b>Apache</b>	<b>Nginx</b>	<b>Lighthttp</b>	<b>Sun Web Server</b>	<b>ISS</b>
Multiplataforma					
Software Libre					
Balanceo de carga					
Tolerancia a fallos					
Compatible con IPV6					
Lenguajes de programación que soporta					
Porcentaje de utilización en el mercado					
Consumo de Recursos					

## ANEXO 3

### FORMATO DE RECOPIACIÓN DE DATOS

**Datos a Recopilar:** Características y especificaciones de las distribuciones Linux.

**Objetivo:** Recopilar información e identificar las Características y especificaciones de las distribuciones Linux como plataforma para los servidores web.

Tabla 23.

	<b>Ubuntu Server 14.04 LTS</b>	<b>Ubuntu Server 16.04 LTS</b>	<b>Red Hat Enterprise</b>
Versión del kernel			
Estado de la distribución			

## ANEXO 4

### GUÍA DE OBSERVACIÓN

**Situación a Observar:** Comportamiento de los servidores web.

**Objetivo:** Identificar y documentar los valores o lecturas obtenidas de la experimentación en cada uno de los servidores web evaluados.

Tabla 24.

		<b>Número de Pruebas Realizadas</b>	<b>Valor Obtenido</b>
Consumo	de		
CPU			
Tiempo	de		
Respuesta			
Consumo	de		
memoria			