

Uso de sockets en la informática: Una Revisión Sistemática de la literatura

Margoth Guaraca¹, Guadalupe Cuascota², Milton Quilisma²
José Jácome², Fernando Imbaquingo³

¹Universidad de las Fuerzas Armadas ESPE, Sede Santo Domingo, Ingeniería en Tecnologías de la Información

²Universidad Técnica del Norte. Facultad de Ingeniería en Ciencias Aplicadas.

² Universidad Técnica del Norte. Facultad de Ciencias Administrativas y Económicas.

³ Universidad Técnica del Norte. La UEmprende. Ibarra, Ecuador

¹*megaraca@espe.edu.ec*, ²*{pgcuascotac@utn.edu.ec, maquilismalp1@utn.edu.ec, jgjacome@utn.edu.ec}*,
³*efimbaquingo@lauemprende.com*

RESUMEN

Los sockets en el área de la informática son una alternativa tecnológica planteada como mecanismo de comunicación eficiente entre aplicaciones que trabajan bajo la arquitectura cliente-servidor, se pueden aplicar en diferentes áreas relacionadas con la informática. El presente artículo hace referencia a una revisión sistemática de literatura (SLR) con el fin de obtener información importante de los diferentes usos o aplicaciones de sockets en las áreas informáticas para su posterior análisis. Para ello se realiza la selección de fuentes bibliográficas confiables en donde se obtuvo información relevante de artículos en revistas científicas validadas con un factor de impacto de calidad, obteniendo como resultado que el uso de sockets mejora el flujo de información y al ser aplicados en las diferentes áreas tecnológicas como: Inteligencia artificial, seguridad informática, arquitecturas de software, servicios web, sistemas operativos y en las redes de comunicación influyen de manera positiva mejorando su rendimiento, todo esto es descrito en una matriz de aplicaciones de la misma forma se elaboró una matriz de ventajas de uso, la cual permitió concluir que los sockets son versátiles y viables para ser aplicados en las diferentes áreas tecnológicas además de generar una serie de ventajas como mejora del rendimiento, transmisión de datos, eliminación de cuello de botella.

Keywords: informática, rendimiento, aplicaciones, protocolo, arquitectura de software, comunicación, sockets, MuniSocket

Introducción

El avance tecnológico ha llevado a los investigadores a realizar diferentes estudios para mantener a los usuarios siempre conectados de manera rápida y eficiente, para ello se busca continuamente soluciones factibles que ayuden a mejorar el proceso de desarrollo de software y la flexibilidad de los sistemas para que se adapten a las demandas de los usuarios [1],[2], en consecuencia, se emplean diferentes tecnologías tanto para el desarrollo de software, integración de servicios como para el aumento de seguridad y control en las redes sobre las cuales se usa las aplicaciones [3]. El conocimiento de estas tecnologías que se encuentran a la vanguardia conlleva saber cómo aplicarlas y que se pretende solucionar debido a la diversidad de áreas en las cuales se puede lograr mejoras notables para beneficio de los interesados.

En el mundo tecnológico es fundamental tener un nivel de prestación de servicios adecuado en los sistemas informáticos, para llegar a obtener esto, una de las soluciones tecnológicas más nombradas es la implementación de sistemas distribuidos [4] y como componente de comunicación se encuentran los sockets los cuales proporcionan múltiples ventajas tales como: aumento de escalabilidad, mejora de rendimiento, elimina cuellos de botella, comunicación eficiente y seguridad, contribuyendo de manera exponencial al mejoramiento del sistema o software [5],[6].

En cuanto a seguridad en los sistemas informáticos se implementa protocolos o certificados como Secure Socket Layer (SSL) que con técnicas criptográficas contribuyen a mantener datos seguros [7],[8]. En medios electrónicos el paso de información es cotidiano por ello es indispensable que los datos lleguen a su destino, con un alto tráfico en la red los sockets reducen la latencia y hace que el flujo de la información sea constante evitando pérdidas de información [9]. El protocolo de comunicación puede ser Transmission Control Protocol (TCP) o User Datagram Protocol (UDP).



El objetivo de este trabajo es realizar una revisión sistemática de la literatura para obtener suficiente información de las aplicaciones de sockets en las diferentes disciplinas de la informática, ventajas de uso de esta tecnología y una conceptualización general. Todo ello detallada en el presente artículo en el que también se menciona el proceso utilizado para la obtención de información relevante además el resultado en una matriz de ventajas de uso de sockets considerado por varios autores y finalmente una conclusión formulada en relación con los resultados obtenidos.

Metodología

Para la presente investigación se ha seleccionado revisión sistemática de la literatura (SLR) con el fin de sintetizar la información existente sobre el área temática [10] "sockets" del cual se obtiene información relevante para su posterior análisis. Todo ello es descrito en un proceso como se muestra en la Figura 1 que consta de 4 fases: (i) preguntas de investigación, (ii) búsqueda de documentos, (iii) selección de artículos, (iv) extracción de datos relevantes. Cada una de estas fases se describen a continuación:

Preguntas de investigación

Se establecieron 2 preguntas de investigación las cuales permitieron indagar sobre el tema de estudio, en esta investigación se basa en los diferentes usos o aplicaciones que se le da a un socket en los campos o áreas de estudio relacionados con la informática. Para ello se establecieron las siguientes preguntas:

Tabla 1: Preguntas de investigación (PI)

Número	Preguntas de investigación	Motivación
PI1	¿Cuáles son los usos o aplicaciones que se da a los sockets en las diferentes ramas de la informática?	Identificar los usos que se les da a los sockets en las diferentes ramas de la informática
PI2	¿Cuáles son los beneficios o ventajas que aportan los sockets?	Identificar los beneficios del uso de los sockets.

Búsqueda de documentos

Este proceso inicia con la selección de base de datos bibliográficas considerando el tema de interés una de ellas es IEEE Xplore [11], Sciece Direct[12], SpringerLink [13] Taylor & Francis [14] y Microsoft Academic [15], que son fuentes digitales que contienen información del área de investigación,. Una vez seleccionadas las distintas librerías digitales identificamos las palabras claves (("sockets" OR "sockets programing") AND ("sockets in distributed systems")) como cadenas de búsqueda del cual se obtiene un listado de documentos recuperados, la Tabla 2 muestra la cadena de búsqueda utilizada en cada librería digital.

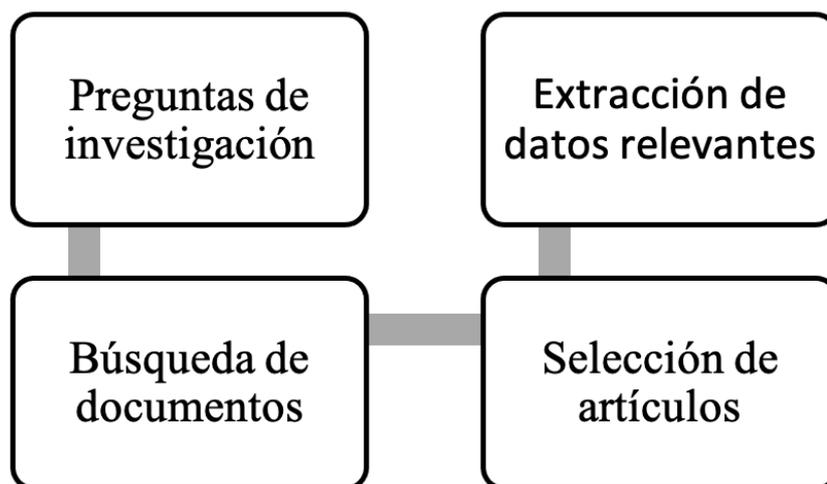


Figura 1. Diagrama del proceso de investigación SLR

Tabla 2: Cadenas de búsqueda utilizadas en las librerías digitales

Criterios	IEEE Xplore	SpringerLink	Science Direct	Microsoft Academic	Taylor & Francis
Cadenas de búsqueda	(Sockets)	(sockets sockets programing) and (sockets)	(sockets in distributed systems)	(sockets in distributed systems)	(sockets sockets programing or sockets) and (sockets in distributed systems)

Selección de artículos

Para la selección de artículos se consideró la totalidad de documentos recuperados entre (i) libros, (ii) capítulos de libro, (iii) revistas, (iv) conferencias y (v) patentes de cada una de las librerías a partir del cual se empiezan las 3 fases de selección. En la primera fase se aplicaron criterios de exclusión e inclusión. Los criterios de inclusión que se establecieron fueron la aplicación del filtro considerando únicamente información de (i) libros y (ii) artículos. Todos estos pertenecientes a la disciplina de ciencias de la computación e ingeniería así también el año de publicación validada entre el rango de los últimos 5 años incluido el 2021.

Los criterios de exclusión fueron información tanto de (i) conferencias como (ii) patentes y (iii) capítulos de libros, así también información perteneciente al área de la medicina ya que la mayor parte de resultados de búsqueda se relacionaban a esta área, obteniendo un total de resultados de la selección primaria como se muestra en la Tabla 3.

En la segunda fase se considera las preguntas de investigación planteadas, para ello se revisó inicialmente el título y el resumen

Finalmente, se validó el factor de impacto de las revistas científicas, en el portal Scimago Journal & Country Rank (SJR) [16] que permitió validar revistas con mayor índice de calidad. Para ello se buscó el cuartil al cual pertenece, considerando únicamente los pertenecientes all Q1, Q2, y Q3, ya que estos son indicadores de mayor importancia en todo el rango de revistas del área, obteniendo un total de resultados mostrados en la selección final (Tabla 3), obteniendo un total de 75 documentos al aplicar las fases de selección.



Tabla 3: Selección de artículos

Librería digital	Total de resultados encontrado	Selección primaria	Selección final
IEEE	4648	388	24
Microsoft Academic	1000	731	14
Springer Link	47000	26282	20
Science Direct	15009	30	15
Taylor & Francis	23	10	2
Total	89782	27641	75

Extracción de datos relevantes

De los 75 trabajos seleccionados se revisaron tanto el título, resumen, introducción y resultados, para constar que contengan información relacionada a las preguntas de investigación para el posterior análisis e interpretación de resultados obteniendo un total de 40 artículos preseleccionados. De esos se realizó una revisión y selección más detallada obteniendo un total de 25 artículos distribuidos, 16 en la Tabla 7 y 7 en la Tabla 8.

Los datos extraídos constan principalmente la conceptualización de sockets, el funcionamiento y la estructura establecida para la transferencia de datos mediante sockets. Además, considerando las preguntas de investigación se identificaron las áreas en las que se aplica los sockets y los beneficios de este aporte en su aplicación.

Resultados

Los sockets se definen generalmente como un método de comunicación de dos vías entre dos programas que se ejecutan a través de la red [17] y utiliza una arquitectura cliente-servidor, donde el cliente realiza peticiones de conexión al servidor para luego el servidor aceptar y establecer la conexión [18], para establecer la comunicación es necesario 3 componentes principales que se muestra en la Tabla 4.

Tabla 4. Componentes de los sockets (Elaboración propia)

Componente	Definición
Dirección IP	Dentro de la red los hosts deben estar identificados para establecer conexión
Puerto	Punto por el cual los hosts van a establecer la conexión.
Protocolo	Modelo de conexión para transmitir la información.

Los tres componentes mencionados en la Tabla 4 son muy importantes ya que si uno de ellos no se encuentra bien establecido la conexión y el socket no funciona. La dirección IP debe ser la adecuada dentro de una red, el puerto el cual es un número no debe ser de los más utilizados debido a que puede ser fácilmente víctima de ataques, y por último el protocolo que puede ser TCP o UDP (Tabla 5), en la utilización de los sockets es importante determinar cuál de ellos se va a usar debido a que cada protocolo realiza la conexión y el paso de la información de manera diferente.

Tabla 5. Protocolos de comunicación (Elaboración propia)

TCP	UDP
Es un modelo para la comunicación utilizada en áreas de redes locales o internet para la transferencia de datos [19], en sockets este protocolo permite la transmisión de flujo de datos que garantiza el envío seguro de información y que el tiempo de respuesta del servidor sea más corto es un modelo de transmisión con conexión [20].	Es un protocolo de datagramas de usuario que usa un modelo de transmisión sin conexión haciendo que el tráfico de datos sea susceptible a pérdida de la información [21] volviéndose más vulnerable a cualquier tipo de ataques afectando el rendimiento de la red [22].

De las diferentes estructuras que se le puede dar a un socket estas cuentan con funciones principales definidas en cualquier tipo [23]. Las funciones descritas en la Tabla 6 se establece como un ciclo de vida del socket descrito por la mayoría de los autores y es representado como un algoritmo el que cada paso cumple una determinada función. En una capa de sockets a nivel de usuario sobre la arquitectura de interfaz virtual SOVIA [24], la gestión empieza estableciendo conexión para el cual se habilita una canal denominado socket el servidor escucha la solicitud de petición y acepta la solicitud se realiza el proceso de transferencia y finalmente se cierra la conexión usando cinco tipos de paquetes para cerrar completamente la sesión.

Tabla 6. Matriz de funciones(Elaboración propia) basada en [25]

Función	Descripción
socket();	Crea una sesión, donde se describe la información para la conexión
connect();	Establece una conexión ejecutando el modo cliente y espera una respuesta
accept();	Acepta la conexión del cliente y ejecuta el servidor dando respuesta
send(); write();	Envío de mensajes confiables durante el proceso de sesión del socket
recive(); read();	Recepción de mensajes confiables durante el proceso de sesión del socket
closeSocket();	Cierra la conexión y termina una sesión abierta

PI 1 ¿Cuáles son los usos o aplicaciones que se da a los sockets en las diferentes ramas de la informática?

El uso variado de los sockets en las tecnologías promueve una mejora en el flujo de la información[26], al ser aplicable en diferentes áreas tecnológicas contribuye para llegar a obtener tecnologías robustas.

En la Tabla 7 se muestra diferentes áreas y los usos o aplicaciones sobre los que se puede trabajar con sockets, se ha tomado diferentes estudios que han causado impacto en el área sobre el que se han empleado

Tabla 7. Matriz de aplicaciones (Elaboración propia)

Áreas Informática	Uso/Aplicación	Autor/es	Base de datos bibliográfica	Año	País
INTELIGENCIA ARTIFICIAL	Cuando se requiere una gran potencia informática para procesamiento de información en redes neuronales es aplicable socket java logrando una solución informática paralela [27].	Gursoy Osman Sharif Md. Haidar	IEEE Xplore	2018	Macedonia
SEGURIDAD	Los ataques cibernéticos en internet es un hecho notable, mediante una capa segura de conexión SSL que utiliza autenticación del cliente se obtiene un canal cliente- servidor con mayor nivel de integridad[28].	Bhatt D.V. Schulze S. Hancke G.P.	IEEE Xplore	2006	South Africa
	Se propone una técnica, llamada tamiz de seguridad, que mejora el rendimiento de la transmisión de documentos basada en SSL[29]	Lim Norman Majumdar Shikharesh Srivastava Vineet	Taylor & Francis	2015	Canadá
	En el mercado en línea es importante la comunicación entre la empresa y el cliente, una capa segura de sockets contribuye un mayor rendimiento y mejor control en los accesos de los clientes[8]	Weaver A.C.	IEEE Xplore	2006	USA
WEB	Actualizaciones de HTML permite la implementación de sockets web llegando a que las aplicaciones reciban actualizaciones asíncronas. Los sockets Web permiten una baja latencia, aumento en la escalabilidad dando mayor calidad en la Web[30].	Qveflander, Nikolai	SpringerLink	2010	USA
	Mediante la implementación de Websockets se mejora el trabajo compartido en donde los usuarios pueden acceder de manera remota desde cualquier navegador que soporte Websockets permitiendo guardar sesiones [31]	Xavier Sotomayo Diego Ordóñez Camacho	Scielo	2017	Chile
	Sistema iterativo de respuesta estudiantil utilizando iBeacon y websocket en el que una aplicación y la interfaz de un navegador interactúan, con el fin de proporcionar un aprendizaje en las aulas de clase. Para ello se hace uso de la tecnología websocket que permite crear conexiones entre el cliente y el servidor [32].	Hung-Hsu Tsai ¹ Chih-Tsan Chang ² Xin-Yu Hou ¹ You-Ming Yong ¹ Kuo-Ching Chiou ³ Pao-Ta Yu	SpringerLink	2019	California

SISTEMAS OPERATIVOS	El método, denominado out-sourcing de socket, reemplaza la capa de socket en un sistema operativo invitado (SO), la sub-contratación de sockets aumenta el rendimiento al eliminar la duplicidad de mensajes entre cliente y servidor[33].	Eiraku Hideki Shinjo Yasushi Pu Calton Koh Young-gyun Kato Kazuhiko	CiteSeer	2007	Japón
	La utilización de sockets Netlink en Linux es un sistema de comunicación flexible entre el espacio del usuario y el núcleo siendo un buen método de comunicación de procesos[34]	Neira Pablo Gasca Rafael M. Lefevre Laurent	InterScience	2010	Francia
ARQUITECTURA DE APLICACIONES	Una arquitectura con la estructura diseñada para lograr una mayor estandarización, flexibilidad, integración y agilidad en el proceso de mantenimiento, está implementada por componentes tecnológicos reutilizables como sockets[1].	Gómez Oiner Machado Alleiny Cabrera Alexis	Google Scholar	2017	Venezuela
	Implementación de una capa de sockets a nivel de usuario sobre una arquitectura de interfaz virtual, duplicando el ancho de banda de transferencia de archivos en FTP para acelerar el rendimiento de las aplicaciones[35].	Kim Jin Soo Kim Kangho Jung Sung In Ha Soonhoi	IEEE Xplore	2018	India



REDES DE COMPUTADORAS	Se evalúa el rendimiento del sistema de comunicación socket basado en PCIe mediante la medición del consumo de ancho de banda al enviar y recibir datos utilizando el protocolo TCP/IP[36].	Shim Cheol Shinde Rupali Choi Min	SpringerLink	2019	Correa del Sur
	Se presenta la técnica MuniSocket en redes duales para el equilibrio de carga de mensajes grandes para mejorar el tiempo de transferencia, donde los mensajes del usuario son fragmentados para luego ser enviados por canales multihilo concurrentes[5].	Mohamed Nader Al-Jaroodi Jameela Jiang Hong	Science Direct	2005	USA
	Implementación de un nuevo tipo de socket denominado UDTCP, para la transmisión de archivos de multimedia, no son adecuados para transmisión de datos sensibles, pero es tolerante a errores [37]	S.Y. Wang	IEEE Xplore	2006	USA
	Generación de un número adecuado de sockets TCP en transferencias de datos de forma paralela con el fin de que la distribución de carga sea equitativa para que se mejore el rendimiento y no se genere el cuello de botella. [38]	Eitan Altman, Dhiman Barman Bruno Tuffin Milan Vojnovic	IEEE Xplore	2006	USA
	Active Network Socket Programming (ANSP) similar a la programación de sockets para elaborar programas de red, este modelo es fácil de implementar en cualquier aplicación y es útil cuando se requiere la comunicación dentro de una red heterogénea de una aplicación en otra región de la red además con esto se puede monitorear el rendimiento de la red[39].	Malik Kamran Khan Osama Mobashir Tahir Sarwar Mansoor	Science Direct	2004	Taiwan

Esta lista de trabajos realizados muestra algunas de las aplicaciones que tienen los sockets, varios de los autores mencionan a los sockets como un buen componente para la transferencia de información que se puede implementar.

¿Cuáles son los beneficios o ventajas que aporta los sockets?

Entre la búsqueda relacionada a ventajas de uso de sockets en las diferentes áreas se encuentran algunos conceptos como mayor rendimiento(MR), incremento de la escalabilidad(IE), elimina cuellos de botella(ECB), comunicación eficiente(CE), mejora la seguridad(S) y Fiabilidad(F), que destacan varios autores como se muestra en la Tabla 8.

Tabla 8. Ventajas de uso de sockets. (Elaboración propia)

ARTÍCULOS	CONCEPTOS					
	M R	I E	ECB	C E	S	F
Protocolo de sockets directo (SDP) mediante el modelo cero copias, transfieren datos entre los buffers de la aplicación [40].	X		X	X		
Los sockets aplicados en laboratorios virtual de red basados en algoritmos de aprendizaje automático bajo el protocolo TCP/IP para la transmisión de mensajes [9]					X	X
La comunicación del procesado mediante un socket sobresaturado degrada el rendimiento, pero este componente permite la escalabilidad mediante la implementación de otros sockets interconectados sobre un mismo puerto disminuyendo el tiempo de ejecución de las tareas.[21]	X	X	X			
Java Fast Sockets elimina la limitación de los sockets tradicionales de sobrecargas en clústers de alto rendimiento, su interoperabilidad permite su fácil implementación[41]	X		X			X
Una variante de los sockets denominados sockets migratorios, su vida útil es la duración del proceso migratorio donde estos sockets mitigan la sobrecarga de migración de procesos en los clústers reduciendo considerablemente este problema de sobrecarga en comparación con otras soluciones[42]	X			X		
Un modelo de programación adaptado a servidores web multinúcleo que reduce el tiempo de respuesta de las solicitudes mediante la implementación de multi sockets [43].	X		X			X
Capa de conexión segura con sockets (SSL), para proporcionar un canal seguro en el intercambio de información en conexiones inalámbricas de sensores mediante la criptografía de ADN que cifra la información para que se mantenga oculta [44]					X	
Arquitectura de sistema de procesador de seguridad en el que se implementa una capa de conexión segura SSL y IPSec de Gbps obteniendo una serie de ventajas en la transferencia de datos [45].				X	X	
Implementación Stack 6 Via Linux utilizando la tecnología sockets mediante el modo de datagrama, secuencia y sin formato para la indicación externa [46]	X					X

Los sockets son una solución tecnológica ideal para cualquier tipo de uso sin embargo los sockets UDP cuentan con algunas desventajas tales como la pérdida de la información y no asegura la confidencialidad de la información por ello la mayor cantidad de investigaciones son realizadas bajo el protocolo TCP.



Conclusiones

En el mundo tecnológico actual es indispensable trabajar con sistemas y aplicaciones de las diferentes áreas de la informática (Tabla 8) con un mayor nivel de disponibilidad y rendimiento para esto el uso de sockets contribuye de manera notable debido a que trabaja distribuyendo cargas en el manejo de la información o conexión entre usuarios.

Los sockets son implementados en varias ramas de la informática, especialmente en redes de computadores, son componente de comunicación entre varios equipos o procesos que están dentro de una red, haciendo de esta herramienta más adaptable a esta área con la finalidad de mejorar el rendimiento y la transmisión de datos (Tabla 7).

Con la información encontrada en la revisión de literatura se muestra un cuadro de resultados de ventajas de uso de sockets el cual expone que varios autores coinciden que los sockets implementados en el intercambio de flujo de datos generan una comunicación eficiente (Tabla 8).

Además, se puede observar en los usos/aplicaciones descritos en la Tabla 7, indican que en las aplicaciones que utilizan sockets y transferencia de archivos de forma paralela hace que la distribución de carga sea equitativa y se eliminen los cuellos de botella y con ello mejore el tiempo de respuesta ante grandes cantidades de solicitudes sin que afecte su rendimiento (Tabla 8).

Referencias

- O. Gómez, A. Machado, and A. Cabrera, "Arquitectura de integración basada en socket para sistemas distribuidos," *Espacios*, vol. 38, no. 59, p. 6, 2017.
- B. Ciubotaru and G.-M. Muntean, *Advanced network programming -- principles and techniques : network application programming with java*. 2013.
- H. Subramoni, F. Petrini, V. Agarwal, and D. Pasetto, "Intra-socket and inter-socket communication in multi-core systems," *IEEE Comput. Archit. Lett.*, vol. 9, no. 1, pp. 13–16, Jan. 2010, doi: 10.1109/L-CA.2010.4.
- J. W. Jang and J. S. Kim, "Design issues and performance comparisons in supporting the sockets interface over user-level communication architecture," *J. Supercomput.*, vol. 39, no. 2, pp. 205–226, Feb. 2007, doi: 10.1007/s11227-007-0109-5.
- N. Mohamed, J. Al-Jaroodi, and H. Jiang, "Dependable user-level socket over dual networks," *J. Parallel Distrib. Comput.*, vol. 65, no. 10, pp. 1261–1270, Oct. 2005, doi: 10.1016/j.jpdc.2005.04.016.
- D. K. Y. Yau and S. S. Lam, "Migrating sockets - End system support for networking with quality of service guarantees," *IEEE/ACM Trans. Netw.*, vol. 6, no. 6, pp. 700–716, 2006, doi: 10.1109/90.748083.
- W. Chou, "Inside SSL: The secure sockets layer protocol," *IT Prof.*, vol. 4, no. 4, pp. 47–52, Jul. 2002, doi: 10.1109/MITP.2002.1046644.
- A. C. Weaver, "Secure Sockets Layer," *Computer* (Long Beach, Calif.), vol. 39, no. 4, pp. 88–90, Apr. 2006, doi: 10.1109/mc.2006.138.

- H. Si, C. Sun, B. Chen, L. Shi, and H. Qiao, "Analysis of Socket Communication Technology Based on Machine Learning Algorithms under TCP/IP Protocol in Network Virtual Laboratory System," *IEEE Access*, vol. 7, pp. 80453–80464, 2019, doi: 10.1109/ACCESS.2019.2923052.
- J. D. Velásquez, "Una guía corta para escribir revisiones sistemáticas de literatura parte 4," *DYNA*, vol. 82, no. 190, pp. 9–12, 2015, doi: 10.15446/dyna.v82n190.49511.
- Institute of Electrical and Electronic, "IEEE Xplore Digital Library," 2020. [Online]. Available: <https://ieeexplore.ieee.org/Xplore/home.jsp>. [Accessed: 30-Jan-2020].
- Science Direct, "ScienceDirect.com | Science, health and medical journals, full text articles and books.," 2020. [Online]. Available: <https://www.sciencedirect.com/>. [Accessed: 30-Jan-2020].
- Springer, "Springer Link," 2020. [Online]. Available: <https://link.springer.com>. [Accessed: 30-Jan-2020].
- Taylor & Francis, "Home | Taylor & Francis Group," 2020. [Online]. Available: <https://taylorandfrancis.com/>. [Accessed: 30-Jan-2020].
- Microsoft Academic, "Home | Microsoft Academic," 2020. [Online]. Available: <https://academic.microsoft.com/home>. [Accessed: 30-Jan-2020].
- Scimago Journal & Country Rank, "Scimago Journal & Country Rank," 2020. [Online]. Available: <https://www.scimagojr.com/>. [Accessed: 30-Jan-2020].
- H. J. Fúquene Ardila, "Implementación Cliente Servidor Mediante Sockets," *Rev. Vínculos*, vol. 8, no. 2, pp. 48–59, 2011, doi: 10.14483/2322939X.4197.
- R. L. R. Maata, R. Cordova, B. Sudramurthy, and A. Halibas, "Design and Implementation of Client-Server Based Application Using Socket Programming in a Distributed Computing Environment," in *2017 IEEE International Conference on Computational Intelligence and Computing Research (ICIC)*, 2017, pp. 1–4, doi: 10.1109/ICIC.2017.8524573.
- L. Xu, X. Yu, Y. Feng, F. Han, J. Hu, and Z. Tari, "Comparative studies of router-based observation schemes for anomaly detection in TCP/UDP networks," *Proc. IEEE Int. Conf. Ind. Technol.*, pp. 1832–1837, 2016, doi: 10.1109/ICIT.2016.7475043.
- M. Schläger, B. Rathke, S. Bodenstern, and A. Wolisz, "Advocating a remote socket architecture for internet access using wireless LANs," *Mob. Networks Appl.*, vol. 6, no. 1, pp. 23–42, Jan. 2001, doi: 10.1023/A:1009857619490.
- S. Blasco, J. Bustos, and D. Rivera, "Detection and Containment Amortization UDP Sockets for Multithreading on Multicore Machines," *IEEE Lat. Am. Trans.*, vol. 14, no. 6, pp. 2853–2856, Jun. 2016, doi: 10.1109/TLA.2016.7555264.
- M. H. Wang, L. W. Chen, P. W. Chi, and C. L. Lei, "SDUDP: A Reliable UDP-Based Transmission Protocol over SDN," *IEEE Access*, vol. 5, pp. 5904–5916, 2017, doi: 10.1109/ACCESS.2017.2693376.

- [P. G. S. Florissi, Y. Yemini, and D. Florissi, "QoSockets: A new extension to the sockets API for end-to-end application QoS management," *Comput. Networks*, vol. 35, no. 1, pp. 57–76, 2001, doi: 10.1016/S1389-1286(00)00150-X.
- Jin-Soo Kim, Kangho Kim, and Sung-In Jung, "SOVIA: a user-level sockets layer over virtual interface architecture," *Int. Conf. Clust. Comput.*, pp. 399–408, 2005, doi: 10.1109/clustr.2001.960006.
- B. Y. L. Kimura, H. C. Guardia, and E. dos S. Moreira, "A Session-based Mobile Socket Layer for Disruption Tolerance on the Internet," *IEEE Trans. Mob. Comput.*, vol. 13, no. 8, pp. 1668–1680, Aug. 2014, doi: 10.1109/TMC.2013.76.
- S. Y. Wang, "Designing and implementing a new type of transport-layer socket: The UDTCP socket case," *Comput. Commun.*, vol. 27, no. 3, pp. 262–272, Feb. 2004, doi: 10.1016/j.comcom.2003.07.002.
- O. Gursoy and M. H. Sharif, "Parallel Computing for Artificial Neural Network Training," *Period. Eng. Nat. Sci.*, vol. 6, no. 1, p. 1, Jan. 2018, doi: 10.21533/pen.v6i1.143.
- D. V. Bhatt, S. Schulze, and G. P. Hancke, "Secure Internet Access to Gateway Using Secure Socket Layer," *IEEE Trans. Instrum. Meas.*, vol. 55, no. 3, pp. 793–800, Jun. 2006, doi: 10.1109/TIM.2005.862009.
- N. Lim, S. Majumdar, and V. Srivastava, "Security sieve: a technique for enhancing the performance of secure sockets layer-based distributed systems," *Int. J. Parallel, Emergent Distrib. Syst.*, vol. 31, no. 5, pp. 481–503, Sep. 2016, doi: 10.1080/17445760.2015.1071367.
- N. Qveflander, "Pushing real time data using HTML5 Web Sockets," Umea University, 2010.
- X. Sotomayor and D. Ordóñez Camacho, "Pizarra Virtual Compartida Websockets; una solución para trabajo remoto colaborativo," *Enfoque UTE*, vol. 8, no. 1, p. 374, Feb. 2017, doi: 10.29019/enfoqueute.v8n1.146.
- H. H. Tsai, C. T. Chang, X. Y. Hou, Y. M. Yong, K. C. Chiou, and P. T. Yu, "Interactive student response system with iBeacon and web-socket for flipped classroom learning," *J. Comput. High. Educ.*, vol. 31, no. 2, pp. 340–361, 2019, doi: 10.1007/s12528-019-09226-x.
- H. Eiraku, Y. Shinjo, C. Pu, Y. Koh, and K. Kato, "Fast networking with socket-outsourcing in hosted virtual machine environments," in *Proceedings of the ACM Symposium on Applied Computing*, 2009, pp. 310–317, doi: 10.1145/1529282.1529350.
- P. Neira-Ayuso, R. M. Gasca, and L. Lefevre, "Communicating between the kernel and user-space in Linux using Netlink sockets," *Softw. Pract. Exp.*, p. n/a-n/a, 2010, doi: 10.1002/spe.981.
- J. S. Kim, K. Kim, S. I. Jung, and S. Ha, "Design and implementation of a user-level sockets layer over virtual interface architecture," *Concurr. Comput. Pract. Exp.*, vol. 15, no. 7–8, pp. 727–749, 2003, doi: 10.1002/cpe.721.

- C. Shim, R. Shinde, and M. Choi, "Compatibility enhancement and performance measurement for socket interface with PCIe interconnections," *Human-centric Comput. Inf. Sci.*, vol. 9, no. 1, p. 18, Dec. 2019, doi: 10.1186/s13673-019-0170-0.
- S. Y. Wang, "Designing and implementing a new type of transport-layer socket: The UDTCP socket case," *Comput. Commun.*, vol. 27, no. 3, pp. 262–272, Feb. 2004, doi: 10.1016/j.comcom.2003.07.002.
- E. Altman, D. Barman, B. Tuffin, and M. Vojnović, "Parallel TCP sockets: Simple model, throughput and validation," *Proc. - IEEE INFOCOM*, vol. 00, no. c, 2006, doi: 10.1109/INFOCOM.2006.104.
- K. L. Eddie Law and R. Leung, "A design and implementation of active network socket programming," in *Microprocessors and Microsystems*, 2003, vol. 27, no. 5–6, pp. 277–284, doi: 10.1016/S0141-9331(03)00026-7.
- D. Goldenberg, M. Kagan, R. Ravid, and M. S. Tsirkin, "Zero copy sockets direct protocol over InfiniBand - Preliminary implementation and performance analysis," *Proc. - Symp. High Perform. Interconnects, Hot Interconnects*, vol. 8, pp. 1–10, 2005, doi: 10.1109/CONNECT.2005.35.
- G. L. Taboada, J. Touriño, and R. Doallo, "Java Fast Sockets: Enabling high-speed Java communications on high performance clusters," *Comput. Commun.*, vol. 31, no. 17, pp. 4049–4059, Nov. 2008, doi: 10.1016/j.comcom.2008.08.012.
- K. Malik, O. Khan, T. Mobashir, and M. Sarwar, "Migratable sockets in cluster computing," *J. Syst. Softw.*, vol. 75, no. 1–2, pp. 171–177, Feb. 2005, doi: 10.1016/j.jss.2004.03.023.
- G. You, X. Wang, and Y. Zhao, "An Adaptive Dynamic Request Scheduling Model for Multi-socket, Multi-core Web Servers," *Arab. J. Sci. Eng.*, vol. 42, no. 2, pp. 751–764, 2017, doi: 10.1007/s13369-016-2350-9.
- S. Upadhyaya, "Secure Communication Using DNA Cryptography with Secure Socket Layer (SSL) Protocol in Wireless Sensor Networks," *Procedia Comput. Sci.*, vol. 70, pp. 808–813, 2015, doi: 10.1016/j.procs.2015.10.121.
- H. Wang, G. Bai, and H. Chen, "A Gbps IPsec SSL security processor design and implementation in an FPGA prototyping platform," *J. Signal Process. Syst.*, vol. 58, no. 3, pp. 311–324, 2010, doi: 10.1007/s11265-009-0371-2.
- D. A. Zaitsev and K. M.A., "Implementing Stack E6 via OS Linux Sockets," *J. Adv. Comput. Sci. Technol.*, vol. 1, no. 3, pp. 116–133, 2012, doi: 10.14419/jacst.v1i3.274.

