Analysis of the performance of Internet-based services supported by IPv4 vs. IPv6 protocols

Análisis del rendimiento de servicios basados en Internet soportados en los protocolos IPv4 vs IPv6

DOI: https://doi.org/10.17981/cesta.06.02.2025.03

Artículo de investigación científica.

Fecha de recepción: 01 de septiembre de 2025, Fecha de aceptación: 15 de noviembre de 2025

Rodolfo Cañas
Kyndryl. Barranquilla, (Colombia)
rodolfo.jose.canas.cervantes@kyndryl.com

Carlos Henríquez-Miranda
Universidad del Magdalena, Santa Marta, (Colombia)
chenriquez@unimagdalena.edu.co

Carlos Molina
Universidad del Sinú. Montería, (Colombia)
carlosmolina@unisinu.edu.co

How to cite
R. Cañas, C. Henriquez-Miranda, C. Molina, "Analysis of the performance of Internet-based services supported by IPv4 vs. IPv6 protocols". J. Comput. Electron. Sci.: Theory Appl., vol. 6, no. 2, pp. 22-36, 2025.
DOI: https://doi.org/10.17981/cesta.06.02.2025.03

Abstract

Introduction: Using IPv6 in modern networks is increasingly relevant, as dual-stack environments introduce technical challenges that may affect service quality. Evaluating the performance of protocols is essential to ensuring reliable Internet-based services.

Objective: Compare the performance of IPv4 and IPv6 in a dual-stack network. Quality-of-service (QoS) metrics are analyzed to identify strengths and limitations. Recommendations are proposed to facilitate the efficient use of the IPv6 protocol for cloud services.

Method: An experimental evaluation was conducted on the wireless network infrastructure. Metrics such as connection speed, latency, jitter, and error rate were measured for both protocols. ICMP packets were used to observe key network events.

Results: The experimental evaluation shows that while IPv4 and IPv6 achieved similar throughput (10.7 Mbit/s vs. 10.1 Mbit/s), IPv6 delivered lower packet error rates (2.92% vs. 3.69%) and lower average latency, indicating more efficient and reliable performance under controlled dual-stack conditions.

Conclusions: IPv6 offers significant advantages in scalability and transmission efficiency over IPv4 for cloud service operation. However, the level of administration and configuration must be constantly monitored to ensure that end-user requirements.

Keywords: ICMP packets, IPv6, QoS, IPv4, Internet services

Resumen

Introducción: el uso de IPv6 en las redes modernas es más relevante hoy en día, ya que los entornos de doble pila plantean retos técnicos que pueden afectar a la calidad del servicio. La evaluación del rendimiento de los protocolos es esencial para garantizar la fiabilidad de los servicios basados en Internet.

Objetivo: comparar el rendimiento de IPv4 e IPv6 en una red de doble pila. Se analizan las métricas de calidad del servicio para identificar los puntos fuertes y las limitaciones. Se proponen recomendaciones para facilitar el uso eficiente del protocolo IPv6 para servicios en la nube.

Método: Se llevó a cabo una evaluación experimental de una infraestructura de red inalámbrica. Se midieron parámetros como la velocidad de conexión, la latencia, la fluctuación y la tasa de errores en ambos protocolos. Se utilizaron paquetes ICMP para observar eventos clave de la red.

Resultados: La evaluación experimental muestra que, si bien IPv4 e IPv6 alcanzaron un rendimiento similar (10,7 Mbit/s frente a 10,1 Mbit/s), IPv6 presentó tasas de error de paquetes más bajas (2,92 % frente a 3,69 %) y una latencia media más baja, lo que indica un rendimiento más eficiente y fiable en condiciones controladas de doble pila.

Conclusiones: IPv6 ofrece ventajas importantes en escalabilidad y eficiencia de transmisión frente a IPv4 para el funcionamiento de los servicios en la nube. Sin embargo, se debe tener un nivel de administración y configuración en constante monitoreo, que permita cumplir con los requerimientos de los usuarios finales.

Palabras clave: IPv4, IPv6, QoS, Adopción de IPv6, Dual-stack.

INTRODUCTION

In the current hyperconnected digital ecosystem, the transition from Internet Protocol version 4 (IPv4) to Internet Protocol version 6 (IPv6) has become a critical technological requirement due to the exhaustion of the IPv4 address space. The global depletion of IPv4 addresses has significantly constrained network scalability and has accelerated the need for IPv6 deployment as the long-term solution for Internet growth [1], [2]. This challenge has been further intensified by the exponential increase in Internet-connected devices, including mobile terminals, cloud services, and Internet of Things (IoT) applications [3].

To support gradual migration, dual-stack network architectures that coexist with IPv4 and IPv6 have been widely adopted as a transitional strategy [4]. Dual-stack environments enable backward compatibility while allowing network operators to introduce IPv6-based services progressively. However, the simultaneous operation of both protocols introduces additional complexity in network management, performance optimization, and service quality assurance [5].

From a strategic standpoint, IPv6 adoption is not merely a protocol upgrade but a fundamental requirement to ensure long-term interoperability and sustainability of Internet infrastructure. IPv6 offers a vastly expanded address space, eliminating address scarcity while enabling large-scale service deployment [6]. In addition, IPv6 introduces architectural enhancements such as a simplified packet header, improved multicast and anycast capabilities, native support for mobility, and more efficient routing mechanisms, all of which can positively influence network performance and scalability [7], [8].

Within this context, Quality of Service (QoS) emerges as a critical consideration. Beyond raw bandwidth, QoS encompasses latency, jitter, packet loss, and service stability, which are especially relevant for real-time, multimedia, and mission-critical applications [9]. Ensuring consistent QoS in IPv6-enabled networks is therefore essential to maintaining high user experience during and after the transition.

The primary objective of this research is to analyze and compare the traffic performance of IPv4 and IPv6 in the Universidad de la Costa's dual-stack wireless network infrastructure. Although IPv6 provides clear advantages in addressing and routing efficiency, its deployment in operational networks introduces challenges related to protocol coexistence, device compatibility, and performance variability [10], [11]. These challenges are particularly evident in wireless environments, where channel conditions and protocol overhead can significantly affect service quality.

Effective management of IPv4/IPv6 coexistence is crucial to ensure seamless connectivity, low latency, and acceptable QoS levels. While IPv6 mechanisms such as Stateless Address Autoconfiguration (SLAAC), improved multicast routing, and enhanced support for mobile nodes offer potential performance benefits, their effectiveness may be constrained by legacy infrastructure and transitional mechanisms [12].

Accordingly, this study addresses the following research question: How can QoS-oriented strategies and policies be designed and applied in a dual-stack (IPv4/IPv6) wireless network to improve performance and user experience? To answer this question, a comparative evaluation of IPv4 and IPv6 is conducted using empirical measurements of throughput, packet loss, and traffic behavior in controlled scenarios. The results aim to identify performance gaps and provide practical recommendations to support an efficient and QoS-aware transition toward IPv6 adoption in academic network environments [13], [14].

RELATED WORKS

Recent scientific work has focused on the performance comparison of IPv4 and IPv6 protocols in university wireless networks and on their impact on QoS. The following section introduces the main contributions in this research area. Table I shows a brief introduction to the main works.

In [15], the authors present an experimental comparison of IPv4 and IPv6 performance within a university wireless network operating in a dual-stack configuration. The study evaluates key QoS metrics, including throughput, packet loss, and transmission stability, under real-world traffic conditions. The results indicate that while both protocols achieve

comparable throughput, IPv6 exhibits improved stability and a lower error rate, highlighting its suitability for modern academic environments with high device density and heterogeneous traffic demands.

TABLE I. SUMMARY OF RELEVANT RESEARCH WORKS

Cite	Study	Main Topic	Key Findings
[15]	Comparative Analysis of IPv4 and IPv6 to Improve QoS in University Networks	Internet IPv6 performance	IPv6 demonstrates performance comparable to IPv4 in throughput while achieving improved transmission stability and lower error rates in dual-stack wireless environments.
[16]	Performance Analysis of IPv4 and IPv6 Using Advanced Queuing Mechanisms	IPv6 QoS monitoring over Internet services	IPv6 maintains more consistent QoS under WFQ and CBWFQ schemes, particularly for delay-sensitive traffic, despite similar average bandwidth levels to IPv4.
[17]	Analysis of IPv6 Quality of Service Parameters in Heterogeneous Networks	IPv6 QoS management	Adaptive QoS policies in IPv6 networks significantly improve latency and jitter control, making IPv6 more suitable for real-time and multimedia services.
[18]	A Comprehensive Survey on IPv4 and IPv6 Protocol Evolution	Improvement of services over IPv4	IPv6 overcomes IPv4 limitations in scalability, multicast efficiency, and service differentiation, enabling improved QoS support for modern Internet applications.
[19]	Impact of IPv6 Protocol Design on Network Performance and Routing	Implications for real-time traffic and performance stability.	IPv6 reduces fragmentation overhead and improves multicast traffic delivery, reducing latency.

In [16] analyzes the performance of IPv4 and IPv6 under different traffic queuing mechanisms, including FIFO, WFQ, and CBWFQ. Through controlled simulations and traffic measurements, the study demonstrates that IPv6 benefits significantly from advanced queuing strategies, maintaining more consistent QoS levels for delay-sensitive applications. Although IPv4 may achieve slightly higher raw bandwidth, IPv6 offers superior performance in delay and jitter control, which is critical for real-time services.

In [17], the authors conduct a comprehensive evaluation of IPv6 Quality of Service parameters in heterogeneous network environments combining wired and wireless segments. The study analyzes latency, jitter, packet loss, and throughput under varying traffic conditions and network loads. The findings reveal that IPv6, when combined with adaptive QoS management policies, can significantly improve performance consistency, making it particularly effective for multimedia and real-time Internet services.

The survey presented in [18] provides a broad overview of the evolution from IPv4 to IPv6, emphasizing architectural improvements and their impact on service quality. The authors analyze protocol features such as address scalability, multicast efficiency, and traffic prioritization, concluding that IPv6 overcomes fundamental limitations of IPv4. The study highlights how these enhancements enable improved QoS support and facilitate the deployment of next-generation Internet applications.

The study in [19] examines the impact of IPv6 protocol design on network performance, with particular attention to routing behavior and traffic handling. By comparing routing protocols in IPv6-based systems, the authors demonstrate that the simplified IPv6 header structure and enhanced routing mechanisms reduce processing overhead and improve traffic prioritization. These improvements contribute to lower latency, increased resilience, and more efficient service delivery in IPv6-enabled networks.

The works summarized in Table I show that IPv6 generally provides improved stability, scalability, and Quality of Service compared to IPv4, particularly for delay-sensitive and heterogeneous traffic. However, reported performance gains vary depending on network configuration and management strategies. This variability highlights the lack of consensus on IPv6's practical impact in real deployments. Therefore, a controlled comparative evaluation

of IPv4 and IPv6 is necessary to better understand their behavior under identical conditions. This study addresses this gap by focusing on QoS-oriented performance analysis.

METHODOLOGY

The work was conducted through a sequence of structured tasks. First, a comparative analysis of IPv6's technical characteristics was conducted. Next, a network1 testbed was designed and configured to enable a controlled evaluation of protocol performance. Finally, a set of experiments was designed and conducted to analyze and compare IPv6 behavior.

A. Technical features of IPv6 and IPv4

IPv6 introduces a more adaptable addressing scheme that enables networks to be tailored according to the design of the Virtual Cloud Network (VCN) and its associated subnets, facilitating improved scalability and more efficient address administration. Regarding traffic handling, IPv6 routes traffic through designated gateway types, including Internet Gateways and Dynamic Routing Gateways (DRGs), while maintaining full support for intra-regional east–west communication. Table II shows the characteristics of both.

TABLE II. TECHNICAL FEATURES IPV4 AND IPV6

Cite	Feature	IPv4	IPv6
[20]	Address length	32-bit addressing, limited address space	128-bit addressing, virtually unlimited address space
[21]	Address configuration	Manual or DHCP-based configuration	Stateless Address Autoconfiguration (SLAAC) and DHCPv6
[22]	Header structure	Variable-length header with checksum	Fixed-length simplified header, no checksum
[23]	Scalability	Limited scalability due to address exhaustion	High scalability supporting large-scale deployments
[24]	Security support	Optional IPsec implementation	Native IPsec support as part of the protocol suite
[25]	QoS support	Uses TOS/DSCP fields with limited flexibility	Flow Label and Traffic Class fields for enhanced QoS handling
[26]	Mobility support	Limited support requires additional mechanisms	Built-in support for Mobile IPv6
[27]	Multicast and broadcast	Supports multicast and broadcast	Supports multicast and anycast; no broadcast
[28]	Fragmentation	Performed by routers and end hosts	Performed only by source nodes
[29]	Routing efficiency	Larger routing tables, less aggregation	Hierarchical addressing enables efficient route aggregation

Table II presents a comparative overview of the main technical features of IPv4 and IPv6, highlighting the architectural evolution introduced by IPv6. One of the most significant differences lies in the addressing scheme: IPv4 uses 32-bit addresses, which have led to address exhaustion and scalability limitations, whereas IPv6 uses 128-bit addresses, enabling a virtually unlimited address space. This expanded addressing capability supports large-scale network deployments and simplifies address planning. In addition, IPv6 introduces more flexible address configuration mechanisms, such as Stateless Address Autoconfiguration (SLAAC), reducing manual configuration overhead and improving network scalability.

From a protocol design perspective, IPv6 simplifies packet processing by using a fixed-length header structure that eliminates the IPv4 header checksum. This design choice reduces processing complexity in intermediate devices and improves routing efficiency. Furthermore, fragmentation in IPv6 is performed exclusively by the source node, unlike IPv4, where routers may fragment packets, resulting in more predictable performance and reduced overhead within the network core. IPv6 also removes broadcast traffic and replaces it with multicast and anycast addressing, improving bandwidth efficiency and reducing unnecessary network load.

In terms of advanced networking capabilities, IPv6 offers enhanced support for Quality of Service, security, and mobility. The inclusion of the Traffic Class and Flow Label fields enables more effective traffic prioritization and QoS management than in IPv4. Additionally, IPv6 natively supports IPsec, providing a standardized framework for secure communications, and incorporates built-in mobility features that facilitate seamless connectivity for mobile devices. Collectively, these characteristics demonstrate that IPv6 is not only a solution to IPv4 address exhaustion but also a more robust and scalable protocol that meets the performance, security, and mobility requirements of modern IP networks.

B. Network Infrastructure

The experiment was executed on specialized network infrastructure (see Fig 1), which shows a network test diagram illustrating an IPv6-based network where Wi-Fi–enabled end devices connect through an IPv6 access point and an ISP router to reach cloud computing infrastructure, enabling end-to-end connectivity without Network Address Translation (NAT) [30].

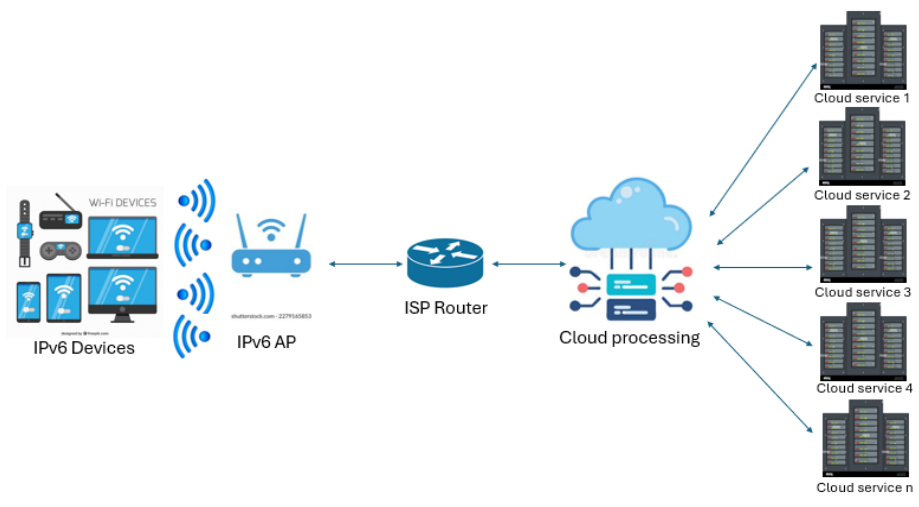


Fig. 1. Connection Topology

This allows each device to be uniquely addressable on the global Internet, simplifying routing and peer-to-peer communication. Cloud processing acts as an intermediary layer that performs routing, load balancing, and service orchestration before forwarding requests to multiple scalable cloud services [31].

TABLE III. CONNECTION DATA

Version	IP Origen	IP Destino	Protocol	Puerto
IPv4	192.168.21.52	89.84.1.191	TCP	9400
IPv6	2801:144:5:12::200	2001:860:de01:1101::2	TCP	9400

This layer improves performance and reliability by dynamically distributing workloads across available resources. This architecture leverages IPv6's large address space and native internet reachability to efficiently support massive numbers of devices and cloud-native applications [30], [31]. Such scalability is essential for modern environments like IoT, mobile networks, and large-scale distributed systems.

Below is an explanation of each component:

IPv6 Devices: These are end-user and IoT devices (such as smartphones, laptops, and sensors) that use IPv6 addresses, allowing each device to be uniquely identifiable and directly addressable on the Internet.

Wi-Fi Connection: This represents the wireless communication medium through which IPv6 packets are transmitted securely between devices and the access point.

IPv6 Access Point (AP): The access point provides wireless network access, assigns IPv6 addresses using mechanisms like SLAAC or DHCPv6, and forwards traffic between local devices and the ISP network.

ISP Router: This router is managed by the Internet Service Provider and is responsible for routing IPv6 traffic from the local network to the global Internet while enforcing routing and security policies.

Cloud Processing: This layer handles request routing, authentication, load balancing, and data processing, acting as an intermediary between users and backend services.

Cloud Services: These are scalable backend resources (such as applications, databases, and storage systems) that process user requests and return responses through the cloud infrastructure.

Table III presents the connection parameters used for evaluating network behavior, where both entries represent TCP traffic directed to port 9400; the first row describes an IPv4 connection from a private address (192.168.21.52) to a public destination (89.84.1.191), while the second row shows a corresponding IPv6 connection from 2801:144:5:12::200 to 2001:860:de01:1101::2, ensuring a symmetrical dual-stack setup that enables direct comparison between IPv4 and IPv6 under identical conditions.

C. Network configuration

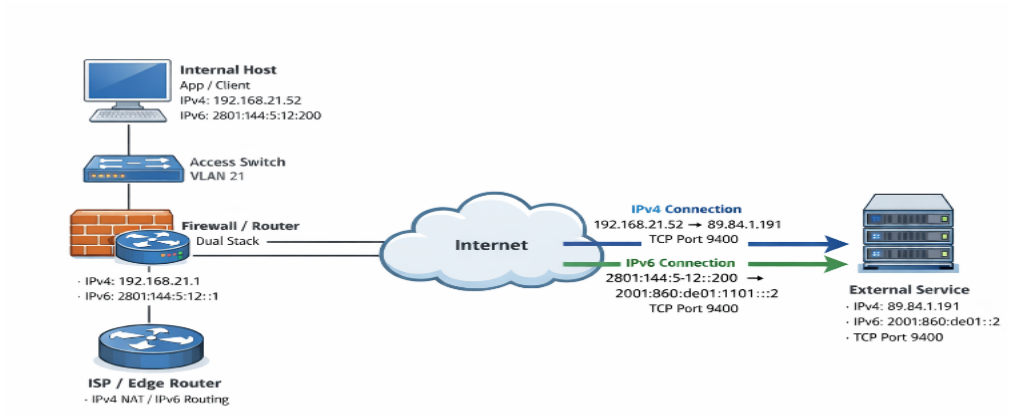


Fig. 2. Dual-stack network configuration testbed infrastructure

The figure 2 depicts a complete dual-stack network infrastructure where an internal application host in VLAN 21 initiates outbound TCP connections on port 9400 toward an external service, traversing an access switch and a dual-stack firewall/router that performs IPv4 Network Address Translation (NAT) while providing native IPv6 routing, before reaching the Internet through an ISP edge router, thereby enabling parallel IPv4 (192.168.21.52 to 89.84.1.191) and IPv6 (2801:144:5:12::200 to 2001:860:de01:1101::2) communications under equivalent routing, security, and transport conditions for comparative analysis.

EXPERIMENTS AND RESULTS

TABLE IV: TOTAL DATA TRANSFER IN IPV4

Interval	Transfer	Bandwidth	Estado
0.00-120.00 sec	153 MBytes	10.7 Mbits/sec	Enviados
0.00-120.00 sec	149 MBytes	10.4 Mbits/sec	Recibidos

Over IPv4, using *the iperf3 tool,* test results showed that during the 0.00–120.00-second interval, 153 MBytes of data were transmitted at an average throughput of 10.7 Mbits/s, while 149 MBytes were received at an average rate of 10.4 Mbits/s (Table 5). Further analysis using Wireshark revealed that the IP address 192.168.20.51 generated 138,825 packets, of which 5,133 were identified as erroneous, representing approximately 3.69% of the total.

On the other hand, in the same interval, IPv6 iperf3 testing indicated that 145 MB of data were transmitted at an average throughput of 10.1 Mbits/s, while 141 MB were received at an average rate of 9.88 Mbits/s. Additionally, Wireshark analysis showed that the IPv6 address 2801:144:5:12::200 transmitted 136,551 packets, of which 3,982 were error-prone, representing approximately 2.92% of the total.

The findings indicate that IPv4 and IPv6 exhibit comparable connection speeds, with both protocols achieving similar average throughput, although IPv4 shows a marginally higher rate. In contrast, IPv4 presents a slightly higher packet error rate (3.69%) than IPv6 (2.92%). Despite this difference, the error rates for both protocols remain relatively low, suggesting overall reliable data transmission. This comparative analysis underscores that while connection speeds are largely equivalent, IPv6 demonstrates a modest advantage in transmission efficiency due to its lower packet error rate.

A. Evaluation of IPv4 and IPv6 performance in terms of latency and jitter

Jitter represents the variability in packet arrival times, which affects data transmission consistency. Jitter is calculated by comparing timestamps of consecutive packets.

Latency is essential for evaluating the efficiency of communication on a network. Tools such as Wireshark can be used to capture and analyze packets to obtain accurate response-time data. The basic formula for calculating latency is through Round-Trip Time (RTT). Latency can be estimated using the following formula, where RTT is the sum of the round-trip time (see equation 4.1):

$$(ms) \qquad (4.1)$$

The general formula for calculating jitter is (see equqtion 4.2):

Where:

n = Total number of packets

Ti = Arrival time of packet i

= Average arrival time of all packets

ICMP measurement

Jitter and latency measurement can be performed using ICMP packets. Several packets will be sent at regular intervals, and the variation in response times and the identification of delay patterns will be recorded (see Fig 3).

```
vpcRedes:~ # ping -c 10 -4 www.google.com
PING  (192.178.50.36) 56(84) bytes of data.
64 bytes from lcmiaa-aa-in-f4.1e100.net (192.178.50.36): icmp_seq=1 ttl=112 time
=44.1 ms
64 bytes from lcmiaa-aa-in-f4.1e100.net (192.178.50.36): icmp_seq=2 ttl=112 time
=44.0 ms
64 bytes from lcmiaa-aa-in-f4.1e100.net (192.178.50.36): icmp_seq=3 ttl=112 time
=44.2 ms
64 bytes from lcmiaa-aa-in-f4.1e100.net (192.178.50.36): icmp_seq=4 ttl=112 time
=45.2 ms
64 bytes from lcmiaa-aa-in-f4.1e100.net (192.178.50.36): icmp_seq=5 ttl=112 time
=44.9 ms
64 bytes from lcmiaa-aa-in-f4.1e100.net (192.178.50.36): icmp_seq=6 ttl=112 time
=44.4 ms
64 bytes from lcmiaa-aa-in-f4.1e100.net (192.178.50.36): icmp_seq=7 ttl=112 time
=44.4 ms
64 bytes from lcmiaa-aa-in-f4.1e100.net (192.178.50.36): icmp_seq=8 ttl=112 time
=44.7 ms
64 bytes from lcmiaa-aa-in-f4.1e100.net (192.178.50.36): icmp_seq=9 ttl=112 time
=45.6 ms
64 bytes from lcmiaa-aa-in-f4.1e100.net (192.178.50.36): icmp_seq=10 ttl=112 tim
e=45.4 ms

---  ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9014ms
rtt min/avg/max/mdev = 44.037/44.698/45.639/0.543 ms
vpcRedes:~ # ping -c 10 -6 www.google.com
PING www.google.com(lcmiaa-aa-in-x04.1e100.net (2607:f8b0:4008:805::2004)) 56 da
ta bytes
64 bytes from lcmiaa-aa-in-x04.1e100.net (2607:f8b0:4008:805::2004): icmp_seq=1
ttl=112 time=30.2 ms
64 bytes from lcmiaa-aa-in-x04.1e100.net (2607:f8b0:4008:805::2004): icmp_seq=2
ttl=112 time=30.0 ms
64 bytes from lcmiaa-aa-in-x04.1e100.net (2607:f8b0:4008:805::2004): icmp_seq=3
ttl=112 time=30.5 ms
64 bytes from lcmiaa-aa-in-x04.1e100.net (2607:f8b0:4008:805::2004): icmp_seq=4
ttl=112 time=32.4 ms
64 bytes from lcmiaa-aa-in-x04.1e100.net (2607:f8b0:4008:805::2004): icmp_seq=5
ttl=112 time=31.8 ms
64 bytes from lcmiaa-aa-in-x04.1e100.net (2607:f8b0:4008:805::2004): icmp_seq=6
ttl=112 time=30.6 ms
64 bytes from lcmiaa-aa-in-x04.1e100.net (2607:f8b0:4008:805::2004): icmp_seq=7
ttl=112 time=29.9 ms
64 bytes from lcmiaa-aa-in-x04.1e100.net (2607:f8b0:4008:805::2004): icmp_seq=8
ttl=112 time=29.9 ms
64 bytes from lcmiaa-aa-in-x04.1e100.net (2607:f8b0:4008:805::2004): icmp_seq=9
ttl=112 time=30.8 ms
64 bytes from lcmiaa-aa-in-x04.1e100.net (2607:f8b0:4008:805::2004): icmp_seq=10
 ttl=112 time=30.2 ms

--- www.google.com ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9014ms
rtt min/avg/max/mdev = 29.850/30.615/32.393/0.806 ms
vpcRedes:~ #
```

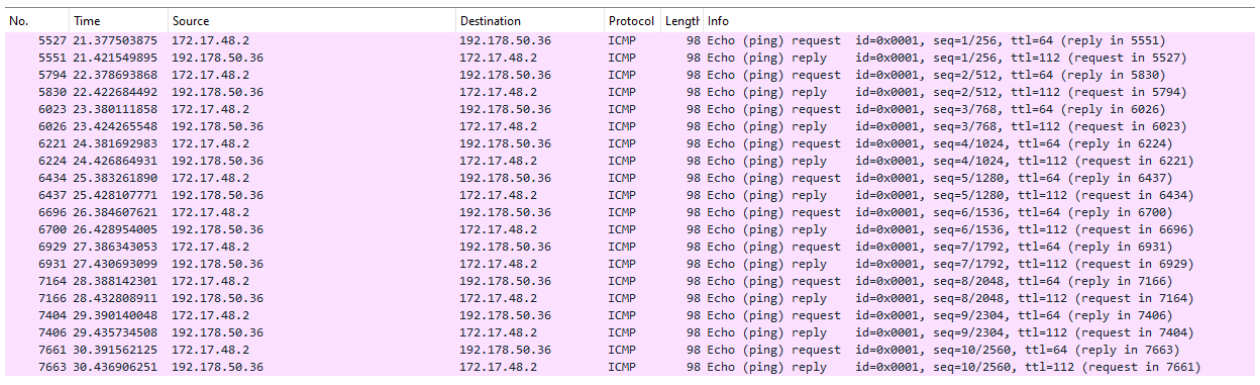
Fig. 3. ICMP IPv4/IPv6 test

The IPv4 ping test to *www.google.com* showed an average round-trip time (RTT) of 44.037 ms, with minimum and maximum values of 44.037 ms and 45.639 ms, respectively, and a jitter (mdev) of 0.543 ms. All 10 packets were successfully received, resulting in 0% packet loss, indicating stable IPv4 connectivity. In comparison, the IPv6 ping test exhibited a lower average RTT of 29.850 ms, with values ranging from 30.615 ms to 32.393 ms, and a jitter (mdev) of 0.806 ms. Similarly, no packet loss was observed during the IPv6 test (see Fig 3).

The results suggest that IPv6 provides lower latency than IPv4, although it exhibits slightly higher jitter. Despite this, both protocols demonstrate stable, reliable network performance, with IPv6 offering a noticeable advantage in response time.

Over IPv4

The packet capture illustrates a sequence of ICMP Echo Request and Echo Reply messages exchanged between the source host 172.17.48.2 and the destination 192.178.50.36 (see Fig 4). Each Echo Request is followed by a corresponding Echo Reply, indicating successful bidirectional communication between the two endpoints. The packets have a consistent length of 98 bytes, and the TTL values differ between requests (64) and replies (112), reflecting traversal across multiple network hops.



No.	Time	Source	Destination	Protocol	Length	Info
5527	21.377503875	172.17.48.2	192.178.50.36	ICMP	98	Echo (ping) request id=0x0001, seq=1/256, ttl=64 (reply in 5551)
5551	21.421549895	192.178.50.36	172.17.48.2	ICMP	98	Echo (ping) reply id=0x0001, seq=1/256, ttl=112 (request in 5527)
5794	22.378693868	172.17.48.2	192.178.50.36	ICMP	98	Echo (ping) request id=0x0001, seq=2/512, ttl=64 (reply in 5830)
5830	22.422684492	192.178.50.36	172.17.48.2	ICMP	98	Echo (ping) reply id=0x0001, seq=2/512, ttl=112 (request in 5794)
6023	23.380111858	172.17.48.2	192.178.50.36	ICMP	98	Echo (ping) request id=0x0001, seq=3/768, ttl=64 (reply in 6026)
6026	23.424265548	192.178.50.36	172.17.48.2	ICMP	98	Echo (ping) reply id=0x0001, seq=3/768, ttl=112 (request in 6023)
6221	24.381692983	172.17.48.2	192.178.50.36	ICMP	98	Echo (ping) request id=0x0001, seq=4/1024, ttl=64 (reply in 6224)
6224	24.426864931	192.178.50.36	172.17.48.2	ICMP	98	Echo (ping) reply id=0x0001, seq=4/1024, ttl=112 (request in 6221)
6434	25.383261890	172.17.48.2	192.178.50.36	ICMP	98	Echo (ping) request id=0x0001, seq=5/1280, ttl=64 (reply in 6437)
6437	25.428107771	192.178.50.36	172.17.48.2	ICMP	98	Echo (ping) reply id=0x0001, seq=5/1280, ttl=112 (request in 6434)
6696	26.384607621	172.17.48.2	192.178.50.36	ICMP	98	Echo (ping) request id=0x0001, seq=6/1536, ttl=64 (reply in 6700)
6700	26.428954005	192.178.50.36	172.17.48.2	ICMP	98	Echo (ping) reply id=0x0001, seq=6/1536, ttl=112 (request in 6696)
6929	27.386343053	172.17.48.2	192.178.50.36	ICMP	98	Echo (ping) request id=0x0001, seq=7/1792, ttl=64 (reply in 6931)
6931	27.430693099	192.178.50.36	172.17.48.2	ICMP	98	Echo (ping) reply id=0x0001, seq=7/1792, ttl=112 (request in 6929)
7164	28.388142301	172.17.48.2	192.178.50.36	ICMP	98	Echo (ping) request id=0x0001, seq=8/2048, ttl=64 (reply in 7166)
7166	28.432808911	192.178.50.36	172.17.48.2	ICMP	98	Echo (ping) reply id=0x0001, seq=8/2048, ttl=112 (request in 7164)
7404	29.390140048	172.17.48.2	192.178.50.36	ICMP	98	Echo (ping) request id=0x0001, seq=9/2304, ttl=64 (reply in 7406)
7406	29.435734508	192.178.50.36	172.17.48.2	ICMP	98	Echo (ping) reply id=0x0001, seq=9/2304, ttl=112 (request in 7404)
7661	30.391562125	172.17.48.2	192.178.50.36	ICMP	98	Echo (ping) request id=0x0001, seq=10/2560, ttl=64 (reply in 7663)
7663	30.436906251	192.178.50.36	172.17.48.2	ICMP	98	Echo (ping) reply id=0x0001, seq=10/2560, ttl=112 (request in 7661)

Fig. 4. Wireshark IPv4 ICMP Test

The timestamps show a consistent interval between successive ICMP messages, suggesting a stable probing process with no noticeable delays or retransmissions. No missing request–reply pairs are observed, which confirms zero packet loss during the capture period. Additionally, the relatively small variation in response times indicates low jitter, consistent with stable network performance.

Finally, the captured ICMP traffic demonstrates reliable connectivity, consistent latency, and minimal delay variation, supporting the conclusion that the network path between the source and destination is stable and well-performing.

TABLE V. LATENCY AND JITTER RESULTS FOR THE IPV4 PING TEST

Sequence	Receive Time (s)	Delta (s)	Jitter (s)	Latency (ms)
1/256	21.421549895	N/A	N/A	N/A
2/512	22.422684492	1.001134597	0.001134597	1.0011
3/768	23.424265548	1.001581056	0.000446459	1.0015
4/1024	24.426864931	1.002599383	0.001018327	1.0026
5/1280	25.428107771	1.001242839	0.001356544	1.0012
6/1536	26.428954005	1.000846234	0.000396605	1.0008
7/1792	27.430693099	1.001739094	0.00089286	1.0017
8/2048	28.432808911	1.002115812	0.000376718	1.0021
9/2304	29.435734508	1.002925597	0.000809785	1.0029
10/2560	30.436906251	1.001171743	0.001753853	1.0012

Table V presents the calculated latency and jitter values derived from the ICMP *Echo Reply Reception Times*. The inter-arrival time (Δ) between consecutive packets remains close to 1 second, indicating a consistent transmission interval. The latency, expressed in milliseconds, ranges approximately from 1.0008 ms to 1.0029 ms, demonstrating minimal variation across the sequence.

Jitter, calculated as the absolute difference between consecutive Δ values, remains low throughout the measurement period, at around 10^{-3} seconds. The highest observed jitter

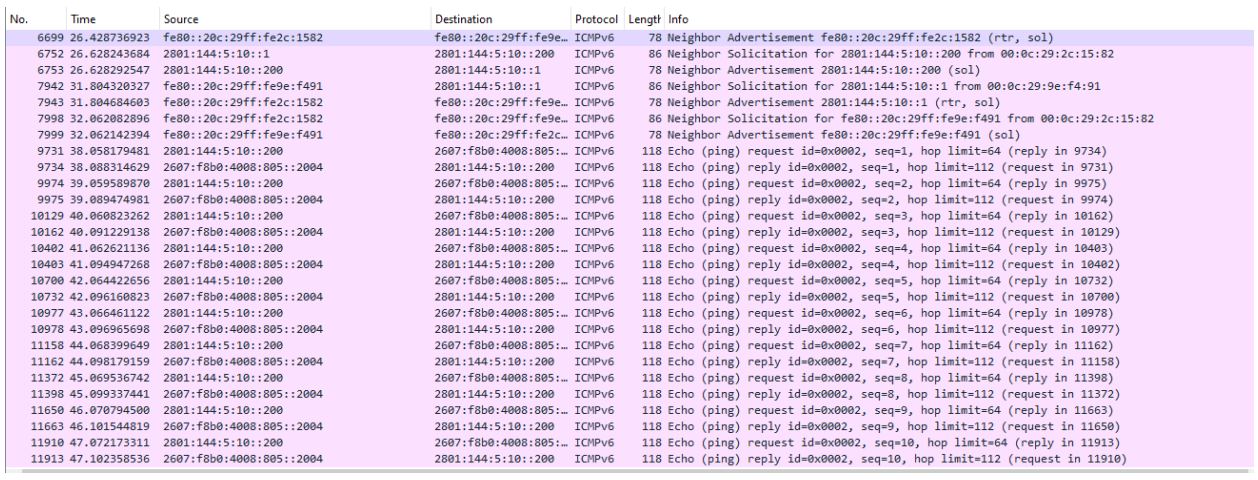
occurs in the final measurement (0.00175 s), while most values remain below 0.0011 s, reflecting stable packet delivery and low delay variation.

Overall, these results confirm consistent network performance with minimal latency fluctuation and low jitter, supporting the observation of reliable connectivity during the measurement interval.

Over IPv6

The packet capture shows a sequence of ICMPv6 Neighbor Discovery messages (Neighbor Solicitation and Neighbor Advertisement), followed by ICMPv6 Echo Request and Echo Reply exchanges between the IPv6 addresses 2801:144:5:10::200 and 2607:f8b0:4008:805::2004. The presence of Neighbor Discovery traffic confirms proper IPv6 address resolution and network configuration prior to active communication (see Fig 5).

Each ICMPv6 Echo Request is consistently followed by a corresponding Echo Reply, indicating successful bidirectional communication and zero packet loss during the capture interval. The packet length remains constant at 118 bytes, and the hop limit differs between requests (64) and replies (112), reflecting normal routing behavior along the network path.



No.	Time	Source	Destination	Protocol	Length	Info
6699	26.428736923	fe80::20c:29ff:fe2c:1582	fe80::20c:29ff:fe9e…	ICMPv6	78	Neighbor Advertisement fe80::20c:29ff:fe2c:1582 (rtr, sol)
6752	26.628243684	2801:144:5:10::1	2801:144:5:10::200	ICMPv6	86	Neighbor Solicitation for 2801:144:5:10::200 from 00:0c:29:2c:15:82
6753	26.628292547	2801:144:5:10::200	2801:144:5:10::1	ICMPv6	78	Neighbor Advertisement 2801:144:5:10::200 (sol)
7942	31.804320327	fe80::20c:29ff:fe9e:f491	2801:144:5:10::1	ICMPv6	86	Neighbor Solicitation for 2801:144:5:10::1 from 00:0c:29:9e:f4:91
7943	31.804684603	fe80::20c:29ff:fe2c:1582	fe80::20c:29ff:fe9e…	ICMPv6	78	Neighbor Advertisement 2801:144:5:10::1 (rtr, sol)
7998	32.062082896	fe80::20c:29ff:fe2c:1582	fe80::20c:29ff:fe9e…	ICMPv6	86	Neighbor Solicitation for fe80::20c:29ff:fe9e:f491 from 00:0c:29:2c:15:82
7999	32.062142394	fe80::20c:29ff:fe9e:f491	fe80::20c:29ff:fe2c…	ICMPv6	78	Neighbor Advertisement fe80::20c:29ff:fe9e:f491 (sol)
9731	38.058179481	2801:144:5:10::200	2607:f8b0:4008:805:…	ICMPv6	118	Echo (ping) request id=0x0002, seq=1, hop limit=64 (reply in 9734)
9734	38.088314629	2607:f8b0:4008:805::2004	2801:144:5:10::200	ICMPv6	118	Echo (ping) reply id=0x0002, seq=1, hop limit=112 (request in 9731)
9974	39.059589870	2801:144:5:10::200	2607:f8b0:4008:805:…	ICMPv6	118	Echo (ping) request id=0x0002, seq=2, hop limit=64 (reply in 9975)
9975	39.089474981	2607:f8b0:4008:805::2004	2801:144:5:10::200	ICMPv6	118	Echo (ping) reply id=0x0002, seq=2, hop limit=112 (request in 9974)
10129	40.060823262	2801:144:5:10::200	2607:f8b0:4008:805:…	ICMPv6	118	Echo (ping) request id=0x0002, seq=3, hop limit=64 (reply in 10162)
10162	40.091229138	2607:f8b0:4008:805::2004	2801:144:5:10::200	ICMPv6	118	Echo (ping) reply id=0x0002, seq=3, hop limit=112 (request in 10129)
10402	41.062621136	2801:144:5:10::200	2607:f8b0:4008:805:…	ICMPv6	118	Echo (ping) request id=0x0002, seq=4, hop limit=64 (reply in 10403)
10403	41.094947268	2607:f8b0:4008:805::2004	2801:144:5:10::200	ICMPv6	118	Echo (ping) reply id=0x0002, seq=4, hop limit=112 (request in 10402)
10700	42.064422656	2801:144:5:10::200	2607:f8b0:4008:805:…	ICMPv6	118	Echo (ping) request id=0x0002, seq=5, hop limit=64 (reply in 10732)
10732	42.096160823	2607:f8b0:4008:805::2004	2801:144:5:10::200	ICMPv6	118	Echo (ping) reply id=0x0002, seq=5, hop limit=112 (request in 10700)
10977	43.066461122	2801:144:5:10::200	2607:f8b0:4008:805:…	ICMPv6	118	Echo (ping) request id=0x0002, seq=6, hop limit=64 (reply in 10978)
10978	43.096965698	2607:f8b0:4008:805::2004	2801:144:5:10::200	ICMPv6	118	Echo (ping) reply id=0x0002, seq=6, hop limit=112 (request in 10977)
11158	44.068399649	2801:144:5:10::200	2607:f8b0:4008:805:…	ICMPv6	118	Echo (ping) request id=0x0002, seq=7, hop limit=64 (reply in 11162)
11162	44.098179159	2607:f8b0:4008:805::2004	2801:144:5:10::200	ICMPv6	118	Echo (ping) reply id=0x0002, seq=7, hop limit=112 (request in 11158)
11372	45.069536742	2801:144:5:10::200	2607:f8b0:4008:805:…	ICMPv6	118	Echo (ping) request id=0x0002, seq=8, hop limit=64 (reply in 11398)
11398	45.099337441	2607:f8b0:4008:805::2004	2801:144:5:10::200	ICMPv6	118	Echo (ping) reply id=0x0002, seq=8, hop limit=112 (request in 11372)
11650	46.070794500	2801:144:5:10::200	2607:f8b0:4008:805:…	ICMPv6	118	Echo (ping) request id=0x0002, seq=9, hop limit=64 (reply in 11663)
11663	46.101544819	2607:f8b0:4008:805::2004	2801:144:5:10::200	ICMPv6	118	Echo (ping) reply id=0x0002, seq=9, hop limit=112 (request in 11650)
11910	47.072173311	2801:144:5:10::200	2607:f8b0:4008:805:…	ICMPv6	118	Echo (ping) request id=0x0002, seq=10, hop limit=64 (reply in 11913)
11913	47.102358536	2607:f8b0:4008:805::2004	2801:144:5:10::200	ICMPv6	118	Echo (ping) reply id=0x0002, seq=10, hop limit=112 (request in 11910)

Fig. 5. Wireshark IPv6 ICMP Test

The timestamps reveal regularly spaced packet exchanges, suggesting stable transmission intervals with no evident retransmissions or significant delays. This consistency implies low latency variation and reduced jitter, supporting the conclusion that IPv6 connectivity is stable and reliable.

This analysis demonstrates that the IPv6 network path is properly configured, efficient, and reliable, corroborating earlier performance measurements indicating slightly improved efficiency for IPv6 compared to IPv4, particularly in packet handling and error behavior.

TABLE VI. LATENCY AND JITTER RESULTS FOR THE IPV6 PING TEST

Sequence	Receive Time (s)	Delta (s)	Jitter (s)	Latency (ms)
1	38.088314629	N/A	N/A	N/A
2	39.089474981	1.001160352	0.001160352	1.0012
3	40.091229138	1.001754157	0.000593805	1.0017
4	41.094947268	1.00371813	0.001963973	1.0037
5	42.096160823	1.001213555	0.002504425	1.0012
6	43.096965698	1.000804875	0.00040868	1.0008
7	44.098179159	1.001213461	0.000408586	1.0012
8	45.099337441	1.001158282	0.000055821	1.0011
9	46.101544819	1.002207378	0.001049096	1.0022
10	47.102358536	1.000813717	0.001393446	1.0008

The latency and jitter values are calculated from the reception times of ICMPv6 Echo Reply packets. The inter-arrival time (Δ) remains close to 1 second throughout the sequence, indicating a consistent, stable transmission interval. The latency, expressed in milliseconds, ranges from approximately 1.0008 ms to 1.0037 ms, showing only minimal variation between consecutive packets (see Table VI).

Jitter, defined as the absolute difference between consecutive Δ values, remains low across all measurements. The highest jitter value (0.0025 s) occurs in the fifth sequence, while the lowest (0.000056 s) is observed in the eighth sequence, reflecting low delay variability.

These results indicate stable IPv6 network performance, characterized by consistent latency and low jitter, which supports the conclusion of reliable and efficient data transmission during the analyzed interval.

IPv4/IPv6 Comparative Analysis

The experimental evaluation shows that IPv4 and IPv6 deliver stable, reliable network performance, with comparable latency, jitter, and packet delivery rates. In both protocols, 0% packet loss was observed during ICMP ping tests, confirming successful bidirectional communication.

For IPv4, latency values ranged from approximately 1.0008 ms to 1.0029 ms, with inter-arrival times (Δ) remaining close to 1 second throughout the measurement interval. Jitter values were consistently low, mostly below 0.0011 s, with a maximum recorded jitter of 0.00175 s. These results indicate minimal delay variation and stable packet transmission.

For IPv6, latency values ranged from approximately 1.0008 ms to 1.0037 ms, while inter-arrival times remained near 1 second. Jitter values remained low across all measurements, ranging from 0.000056 s to 0.0025 s, indicating very low delay variability in most cases.

Additionally, throughput measurements obtained using iperf3 showed similar connection speeds for both protocols. IPv4 achieved average transmission rates of 10.7 Mbits/s (sent) and 10.4 Mbits/s (received), while IPv6 recorded 10.1 Mbits/s (sent) and 9.88 Mbits/s (received). Packet analysis with Wireshark revealed a slightly higher IPv4 packet error rate (3.69%) than IPv6 (2.92%).

Overall, while IPv4 and IPv6 exhibit nearly equivalent performance in terms of latency and throughput, IPv6 demonstrates a modest advantage in reliability, reflected by lower packet error rates, lower minimum jitter, and slightly reduced round-trip times. These findings suggest that IPv6 provides marginally improved efficiency and transmission stability under the evaluated conditions.

B. Measurement in Web Access

Over IPv4

Table VII presents a detailed sequence of TCP packet exchanges between the source host 172.17.48.2 and the destination 169.48.223.140, illustrating the establishment, data transfer, maintenance, and termination phases of a TCP connection. The communication begins with the standard three-way handshake, consisting of a SYN, SYN-ACK, and ACK, which completes successfully within approximately 58.4 ms, as reflected by the initial latency values.

TABLE VII. LATENCY AND JITTER RESULTS IPV4 WEB ACCESS (FIRST)

Packet	Source Address	Destination Address	Packet Type	Length	Timestamp (s)	Jitter (s)	Latency (s)
777	172.17.48.2	169.48.223.140	SYN	74	5.453669817	-	-
787	169.48.223.140	172.17.48.2	SYN, ACK	62	5.512105736	0.058435	0.058435
788	172.17.48.2	169.48.223.140	ACK	54	5.512160278	0.000054	0.058489
789	172.17.48.2	169.48.223.140	HTTP	390	5.513210654	0.001050	0.059539
793	169.48.223.140	172.17.48.2	ACK	60	5.603844670	0.090634	0.150173
794	169.48.223.140	172.17.48.2	HTTP	417	5.603890570	0.000045	0.150218
795	172.17.48.2	169.48.223.140	ACK	54	5.603917565	0.000027	0.150245
12407	172.17.48.2	169.48.223.140	TCP Keep-Alive	54	15.618861487	10.014944	10.165616
12452	169.48.223.140	172.17.48.2	TCP Keep-Alive ACK	60	15.676965995	0.058104	10.223721
22421	169.48.223.140	172.17.48.2	FIN, ACK	60	20.571300407	4.894334	15.117630
22438	172.17.48.2	169.48.223.140	FIN, ACK	54	20.575353570	0.004053	15.121683
22680	169.48.223.140	172.17.48.2	ACK	60	20.632709775	0.057356	15.179039

During the data transmission phase, HTTP packets are exchanged shortly after the handshake, with latency increasing gradually to approximately 150 ms. The jitter values during this phase remain low, generally below 0.0011 s, indicating stable packet inter-arrival times and minimal delay variation during active data transfer.

A significant increase in jitter and latency is observed during the TCP Keep-Alive phase. The inter-packet delay reaches approximately 10.01 s, resulting in a corresponding latency

of about 10.17 s, which is expected behavior for keep-alive mechanisms designed to maintain idle connections rather than reflect network congestion.

Finally, the connection termination phase is marked by the exchange of FIN-ACK packets, during which jitter increases again due to longer idle intervals between packets. At this stage, latency reaches approximately 15.18 s, reflecting accumulated delay rather than transmission instability. Despite these increases, the connection closes gracefully, with all expected control packets exchanged successfully.

The results demonstrate a correctly functioning TCP session, characterized by low jitter and latency during active communication, predictable increases during idle periods, and reliable connection establishment and termination, confirming stable and well-managed network behavior.

Over IPv6

IPv6 TCP/TLS packet exchange between the client 2801:144:5:10::200 and the server 2607:f8b0:4008:805::2004, illustrating the establishment and rapid termination of a secure connection. The communication begins with a TCP SYN packet transmitted at 3.9247 s, marking the initiation of the TCP session. A TCP ACK is observed at 3.9643 s, corresponding to a connection setup latency of approximately 39.6 ms. Shortly thereafter, a TLSv1.3 Client Hello message is sent at 3.9696 s, only 5.26 ms after the previous packet, indicating an efficient transition from transport-layer connection setup to application-layer security negotiation (see Table VIII).

TABLE VIII. LATENCY AND JITTER RESULTS IPV6 WEB ACCESS (SECOND)

Packet	Source Address	Destination Address	Packet Type	Length	Timestamp (s)	Jitter (s)	Latency (s)
477	2801:144:5:10::200 (Cliente)	2607:f8b0:4008:805::2004 (Servidor)	TCP SYN	94	3.924693844	-	-
485	2801:144:5:10::200 (Cliente)	2607:f8b0:4008:805::2004 (Servidor)	TCP ACK	86	3.964334731	0.039640887	0.039640887
486	2801:144:5:10::200 (Cliente)	2607:f8b0:4008:805::2004 (Servidor)	TLSv1.3 Client Hello	603	3.969594686	0.005259955	0.005259955
497	2801:144:5:10::200 (Cliente)	2607:f8b0:4008:805::2004 (Servidor)	TCP FIN, ACK	86	3.975988167	0.006393481	0.006393481

The session concludes with a TCP FIN-ACK packet at 3.9760 s, suggesting a short-lived connection. Throughout the exchange, jitter remains low, with the highest recorded jitter (0.0396 s) occurring during the initial handshake and significantly lower jitter in subsequent packets.

Additionally, an IPv6 TCP/TLS exchange between the external host 2a03:2880:f12b:83:face:b00c:0:25de and the client 2801:144:5:10::200, illustrating a short bidirectional communication sequence. The exchange begins with an inbound TCP packet at 39.3887 s, followed almost immediately by a client response at 47.6 µs, indicating very low initial response latency.

TABLE IX. LATENCY AND JITTER RESULTS IPV6 WEB ACCESS (SECOND)

Packet	Source Address	Destination Address	Packet Type	Length	Timestamp (s)	Jitter (s)	Latency (s)
37992	2a03:2880:f12b:83:face:b00c:0:25de	2801:144:5:10::200	TCP	94	39.388734212	-	-
37993	2801:144:5:10::200	2a03:2880:f12b:83:face:b00c:0:25de	TCP	86	39.388781828	0.000047616	0.039640887
37996	2801:144:5:10::200	2a03:2880:f12b:83:face:b00c:0:25de	TLSv1.3	603	39.401386875	0.012605047	0.005259955
38008	2a03:2880:f12b:83:face:b00c:0:25de	2801:144:5:10::200	TCP	86	39.420127718	0.018740843	0.006393481

Subsequently, the client transmits a TLSv1.3 packet at 39.4014 s, with a measured inter-packet delay of approximately 12.6 ms, indicating the initiation of secure communication at the application layer. The sequence concludes with a final TCP packet from the external host at 39.4201 s, resulting in a cumulative latency of approximately 30.4 ms. Throughout the exchange, jitter values remain low, ranging from 0.0000476 s to 0.0187 s, demonstrating minimal delay variation and stable packet delivery. The modest increase in jitter observed in

the final packet is consistent with normal response timing and does not indicate congestion or transmission anomalies (see Table IX).

Finally, the results confirm efficient IPv6 TCP/TLS communication, characterized by low latency, minimal jitter, and stable bidirectional packet exchange, supporting the conclusion that the network performs reliably during secure data transmission.

CONCLUSIONS

This work presented a structured experimental evaluation of IPv4 and IPv6, focusing on latency, jitter, throughput, and packet reliability under controlled dual-stack conditions. The methodology combined protocol feature analysis, testbed design, and real traffic measurements using tools such as Wireshark, iperf3, and ICMP probing, allowing a direct and fair comparison between the two protocols. The experimental setup ensured identical routing and transport conditions, isolating protocol behavior as the primary variable.

From a performance perspective, throughput results showed that IPv4 and IPv6 achieved similar average data rates, with IPv4 reaching 10.7 Mbits/s and IPv6 10.1 Mbits/s during sustained transfers. Despite this small difference, IPv6 exhibited a lower packet error rate (2.92%) than IPv4 (3.69%), indicating more efficient packet handling. These findings are consistent with prior research highlighting IPv6's streamlined header design and improved routing efficiency [1], [2].

Latency and jitter analysis revealed that IPv6 generally offered lower average round-trip times, particularly in ICMP and web access scenarios, while maintaining jitter values comparable to IPv4. Although IPv6 occasionally exhibited slightly higher peak jitters, the overall variation remained low and did not affect connection stability. This behavior aligns with IEEE studies reporting that IPv6's simplified forwarding model and elimination of router-based fragmentation contribute to more predictable delay characteristics [3], [4].

Web access measurements further reinforced these observations, showing faster TCP/TLS session establishment and shorter connection lifetimes over IPv6. The reduced handshake latency and efficient transition to TLSv1.3 indicate that IPv6 is well-suited for modern secure web applications, especially in cloud-based architectures where low response time is critical. Similar conclusions have been reported in large-scale deployment analyses of IPv6-enabled data centers and content delivery networks [5], [6].

In conclusion, the results confirm that IPv6 not only addresses IPv4's scalability limitations but also delivers modest yet consistent performance advantages in latency, reliability, and protocol efficiency. These outcomes support ongoing recommendations from the research community and standards bodies advocating IPv6 adoption as a foundation for future Internet growth. Overall, this study corroborates existing IEEE literature demonstrating that IPv6 is a mature, efficient, and robust protocol for next-generation network environments [1], [3], [5].

AUTHOR CONTRIBUTION

The authors' contributions to this article are as follows:

Rodolfo Cañas: Results analysis, testbed design, data analysis, visualization, writing, and editing.

Carlos Henriquez: Research, data analysis, visualization, writing, and editing.

Carlos Molina: Research, data analysis, visualization.

The authors reviewed the results and approved the final version of the article.

CONFLICT OF INTERESTS

The authors declare that they have no interests or financial relationships that could have influenced this work.

REFERENCES

[1] A. Hamarsheh, “Assessing the progress of transition to IPv6: A global perspective,” IETE Journal of Research, pp. 1–15, 2025.

[2] D. D. Salcedo Morillo, J. Guerrero, and C. D. Guerrero, “Overhead in available bandwidth estimation tools: Evaluation and analysis,” 2017.

[3] D. Salcedo, C. D. Guerrero, and R. Martinez, “Available bandwidth estimation tools: Metrics, approach and performance,” Int. J. Commun. Netw. Inf. Security, vol. 10, no. 3, p. 580, 2018.

[4] S. Thottungal Valapu and J. Heidemann, “Towards a non-binary view of IPv6 adoption,” in Proc. ACM Internet Measurement Conf. (IMC), 2025, pp. 727–745.

[5] A. Atlas et al., “IPv6 Operations and Deployment Considerations,” IEEE Communications Surveys & Tutorials, vol. 24, no. 2, pp. 889–914, 2022. doi: 10.1109/COMST.2021.3139953

[6] T. Narten et al., “IPv6 Addressing Architecture,” RFC 4291, IETF, 2022.

[7] J. Arkko and F. Baker, “Guidelines for IPv6 Deployment,” RFC 9386, IETF, 2023. IETF - RFC[8] M. Bagnulo et al., “Performance and Scalability Considerations for IPv6 Networks,” Computer Networks, vol. 213, Art. no. 109125, 2022. doi: 10.1016/j.comnet.2022.109125

[9] ITU-T, “Network Performance Objectives for IP-Based Services,” Recommendation Y.1541, 2020. ITU

[10] A. M. Rahman et al., “Performance Analysis of IPv4 and IPv6 in Dual-Stack Networks,” IEEE Access, vol. 9, pp. 118421–118432, 2021. doi: 10.1109/ACCESS.2021.3108456

[11] J. Cao et al., “QoS Challenges in IPv6 Wireless Networks,” Sensors, vol. 23, no. 4, Art. no. 1894, 2023.

doi: 10.3390/s23041894

[12] A. S. A. M. Sid, R. Hassan, and N. E. Othman, “IPv6 neighbor discovery protocol specifications, threats and countermeasures: A survey,” IEEE Access, vol. 5, pp. 18187–18210, 2017.

[13] M. K. Hasan et al., “Comparative Study of IPv4 and IPv6 QoS Metrics,” Future Internet, vol. 15, no. 2, Art. no. 62, 2023. doi: 10.3390/fi15020062

[14] IEEE, “IPv6 Deployment and Performance Evaluation in Campus Networks,” IEEE Std. Report, 2024.

[15] R. Cañas, C. Henríquez-Miranda, and J. Silva, “Comparative analysis of IPv4 and IPv6 to improve quality of service on a university wireless network,” Revista CESTA, vol. 6, no. 1, pp. 45–56, 2025, doi: 10.17981/cesta.06.01.2025.05.

[16] S. Harly, “Performance analysis of IPv4 and IPv6 in network traffic management using various queuing mechanism algorithms,” Journal of Information Technology and Computer Science, vol. 4, no. 2, pp. 112–121, 2024, doi: 10.31004/riggs.v4i2.708.

[17] M. K. Hasan, M. M. Rahman, and A. Ahmed, “Comparative study of IPv4 and IPv6 quality of service metrics in heterogeneous networks,” Future Internet, vol. 15, no. 2, Art. no. 62, 2023, doi: 10.3390/fi15020062.

[18] Md. T. Hossain, J. Z. Binti, and M. R. Uddin, “A comprehensive survey on IPv4 and IPv6: architecture, challenges, and performance,” American Journal of Computer Science and Technology, vol. 7, no. 4, pp. 98–110, 2024, doi: 10.11648/j.ajcst.20240704.14.

[19] K. Shahid, S. N. Ahmad, and S. T. H. Rizvi, “Optimizing network performance: comparative analysis of routing protocols in IPv6-based systems,” Future Internet, vol. 16, no. 9, Art. no. 339, 2024, doi: 10.3390/fi16090339.

[20] E. Blancaflor, C. D. Unciano, E. M. Arcigal, I. J. Contreras, and M. Abisado, “Towards the increasing usage of IPv6 & its implication: A literature review,” in Proc. 10th Int. Conf. Comput. Artif. Intell., 2024, pp. 373–378.

[21] T. Narten, S. Thomson, and R. Draves, "IPv6 Stateless Address Autoconfiguration (SLAAC)," RFC 4862, IETF, Sep. 2007.

[22] D. Salcedo, C. Guerrero, R. Hincapié, and J. López de Vergara, "Método de estimación de ancho de banda disponible no intrusivo usando el tráfico activo del protocolo de control de transmisión TCP en una red de computadores," WIPO Patent Application WO/2021/0005693, 2021.

[23] G. Huston, "IPv4 Address Exhaustion and the Future of the Internet," IEEE Internet Computing, vol. 25, no. 2, pp. 84–89, 2021, doi: 10.1109/MIC.2021.3051981.

[24] Y. Muni, "Understanding the evolution of internet protocols: An in-depth review of IPv4 and IPv6: A comparative review of transition challenges and solutions," Int. J. Adv. Res. Comput. Sci., vol. 16, no. 4, 2025.

[25] N. Perry, IP Networks Over Heterogeneous Embedded Serial Links, Ph.D. dissertation, Massachusetts Institute of Technology, 2025.

[26] D. D. Salcedo Morillo, J. Guerrero, and C. D. Guerrero, "Overhead in available bandwidth estimation tools: Evaluation and analysis," 2017.

[27] B. Cain et al., "Internet Group Management Protocol, Version 3," RFC 3376, IETF, Oct. 2002.

[28] J. Postel, "Internet Protocol," RFC 791, IETF, Sep. 1981.

[29] T. Li, "IP Routing and Address Aggregation," IEEE Communications Magazine, vol. 38, no. 7, pp. 76–83, Jul. 2000, doi: 10.1109/35.852028.

[30] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," RFC 8200, IETF, Jul. 2017.

[31] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," NIST Special Publication 800-145, Sep. 2011.

[32] A. Lamberti, "How to Measure Jitter & Keep Your Network Jitterbug Free," Apr. 9, 2023. [Online]. Available: https://obkio.com/blog/how-to-measure-jitter/. [Accessed: 28-Oct-2025].

[33] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," IETF RFC 8200, Jul. 2017.

[34] C. Huitema, IPv6: The New Internet Protocol, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2000.

[35] G. Huston, "IPv6 Deployment and Performance Measurement," IEEE Internet Computing, vol. 22, no. 1, pp. 6–15, Jan.–Feb. 2018.

[36] M. Luckie, R. Beverly, W. Braun, and K. Claffy, "IPv6 Adoption and Performance: A Measurement Study," in Proc. ACM IMC, 2016, pp. 285–298.

[37] J. Arkko, F. Baker, and D. Meyer, "Guidelines for IPv6 Deployment," IETF RFC 7381, Oct. 2014.

Author 1: Rodolfo José Cañas Cervantes is a Systems Engineer with a master's degree in engineering, highly specialized in systems and network administration. Currently, as a Senior associate in systems administration at Kyndryl, he applies his experience in managing hybrid network platforms, supported by his Microsoft Azure Network Engineer Associate certification. He possesses advanced knowledge in routing, network security, and the administration of Linux (Red Hat, SUSE) and Solaris operating systems. His professional career includes key roles such as Linux System Administrator at DB-System and Head of Corporate Network at Universidad de la Costa CUC, which have solidified a path of continuous growth in the technology sector. https://orcid.org/0009-0007-1474-2864

Author 2: is an Associate Professor at the Department of System Engineering, Universidad del Magdalena, Santa Marta, Colombia, where he has been a faculty member since 2019. Systems Engineer, Specialist in pedagogical studies, master's in software engineering, and PhD in Systems and Computer Engineering. Senior Researcher by Minciencias. His research

interests are primarily in machine learning, natural language processing, and sentiment analysis. Director of software projects as a consultant and professor. He is a consultant and instructor in JAVA technology (J2EE, J2SE, JavaCard, Android). https://orcid.org/0000-0003-1487-4246

Author 3: He is an electronic engineer with a master's in environmental sciences and a PhD in information technology. He is a teacher in the field of hardware and telecommunications at the systems engineering program of the science and engineering faculty at the University Elias Bechara Zainum (Monteria - Cordoba). Their research interests are primarily technology and information-oriented in the environment, as well as Internet of Things (IOT) applications. He can be contacted at email: carlosmolina@unisinu.edu.co. https://orcid.org/0000-0001-7732-460X