



# Implementing And Evaluating Cross-Protocol Communication Between IPV6 And IPV4 Using NAT64

<sup>1</sup>Mohamed Abdullah F, <sup>2</sup>Ikram Mustafa Hussain A, <sup>3</sup>Dhanya P, <sup>4</sup>Chockalingam S RM, <sup>5</sup>Kamaleshwaran S

<sup>1,2,4,5</sup> Student, <sup>3</sup> Assistant Professor

<sup>1</sup>Department of Computer Science and Engineering,

<sup>1</sup>Aalim Muhammed Salegh College of Engineering, Chennai, India

**Abstract:** The transition from IPv4 to IPv6 presents challenges in enabling communication between devices that use both protocols. This project focuses on the implementation and performance evaluation of NAT64 using RIPE Atlas. Using RIPE Atlas probes, we will measure latency, packet loss, and reachability across global networks. NAT64 serves as a solution to facilitate communication by translating IPv6 addresses to IPv4 addresses. Our research will assess the effectiveness of NAT64, identify its limitations, and propose optimizations to enhance cross-protocol communication. NAT64 enables devices that use IPv6 to access IPv4-only resources and vice versa, allowing IPv4 devices to reach IPv6-only services

**Index Terms** - NAT64, IPv6-to-IPv4 Transition, Cross-Protocol Communication.

## I. INTRODUCTION

The migration from IPv4 to IPv6 has become essential due to the depletion of IPv4 address space and the increasing demand for a more scalable and efficient internet infrastructure. IPv6 introduces a vastly larger address space, improved security features, and enhanced network performance. However, IPv4 and IPv6 are inherently incompatible, which poses significant challenges in ensuring seamless communication between networks that rely on different protocols. Many legacy systems, Internet service providers (ISPs), and organizations continue to depend on IPv4, while newer deployments are increasingly adopting IPv6. This co-existence phase creates inter-operability issues, where IPv6-only clients cannot directly communicate with IPv4-only servers, leading to disruptions in connectivity and accessibility. Since a complete and immediate global transition to IPv6 is impractical, interoperability solutions such as NAT64 (Network Address Translation from IPv6 to IPv4) play a pivotal part in bridging the communication gap between these two protocols

NAT64 operates by translating IPv6 addresses into IPv4 addresses and vice versa, allowing IPv6-only clients to access IPv4 resources without requiring dual stack configurations [1]. This method is particularly valuable for ISPs, enterprises, and cloud service providers that need to support IPv6 users while maintaining backward compatibility with the existing IPv4 infrastructure. NAT64 eliminates the need for IPv4 addresses on the client side, making it a viable solution for IPv6 adoption in environments where IPv4 address scarcity is a major concern. However, despite its advantages, NAT64 introduces additional overhead in network translation, which can lead to increased latency, packet loss, and potential service compatibility issues. Since NAT64 modifies the headers of the packet during translation, certain applications—especially those relying on embedded IPv4 addresses or peer-to-peer connections—may experience interruptions. Therefore, a thorough evaluation of the performance, efficiency, and limitations of NAT64 is necessary to determine its viability as a long-term solution for IPv6-IPv4 communication.

This study aims to implement and evaluate NAT64 using RIPE Atlas, a globally distributed measurement platform that enables real-time network performance analysis. By deploying RIPE Atlas probes in different regions, we will measure key network metrics such as latency, packet loss, and reachability when communicating through NAT64 [2]. The results will provide insight into the practical effectiveness of NAT64 under varying network conditions and help identify optimization strategies to improve performance. In addition, we will analyze the impact of NAT64 on real-world applications, highlighting scenarios where it can introduce performance bottlenecks or operational challenges. Through this research, our goal is to contribute to the ongoing discussion on IPv6 transition strategies, offering recommendations on how to improve cross-protocol communication while minimizing disruptions. The findings will help network administrators, service providers, and policy makers make informed decisions about the adoption and optimization of NAT64 in modern networking environments

## II. BACKGROUND

The rapid growth of internet-connected devices and services has led to the exhaustion of IPv4 addresses, necessitating a transition to IPv6, which offers a vastly expanded address space, improved routing efficiency, and enhanced security features. However, IPv4 and IPv6 are inherently incompatible, meaning devices using different protocols cannot communicate directly. This presents a significant challenge during the transition phase, where IPv4-only and IPv6-only networks must coexist and interoperate. While solutions like dual-stack (running both IPv4 and IPv6) are available, they are not always feasible due to increased complexity, cost, and resource requirements. To bridge the communication gap, NAT64 (Network Address Translation from IPv6 to IPv4) has emerged as a key technology that enables IPv6-only clients to access IPv4-only services by translating packet headers between the two protocols. NAT64 is particularly useful in environments such as mobile networks and data centers, where IPv6 adoption is accelerating but compatibility with existing IPv4 infrastructure [3] must be maintained. Compared to other mechanisms like 6 to 4 tunneling or 464XLAT, NAT64 offers a more scalable and efficient solution for cross-protocol communication. Understanding and evaluating NAT64's performance is essential for ensuring reliable and seamless connectivity in mixed-protocol networks as the global internet continues its gradual shift toward full IPv6 adoption.

The depletion of IPv4 addresses has been a growing concern due to the rapid expansion of the internet and the increasing number of connected devices. IPv4, which uses a 32-bit address space, can accommodate approximately 4.3 billion unique addresses, but this number is insufficient to support the exponential growth of modern networks. To address this limitation, IPv6 was introduced with a 128-bit address space, providing an almost unlimited number of IP addresses and enabling improved routing efficiency, enhanced security, and better support for mobile networks. Despite these advantages, IPv6 adoption has been slow and many networks still rely heavily on IPv4 infrastructure. This creates a significant challenge for interoperability, as IPv4 and IPv6 are fundamentally incompatible and cannot communicate directly without specialized transition mechanisms.

Several solutions have been proposed to facilitate the transition from IPv4 to IPv6, including dual stack, tunneling, and translation mechanisms. Dual stack allows devices to support both IPv4 and IPv6 simultaneously, but requires extensive infrastructure upgrades and is not always feasible. Tunneling methods, such as 6to4 and Teredo, encapsulate IPv6 packets within IPv4 packets, but introduce additional overhead and potential performance issues. NAT64 (Network Address Translation from IPv6 to IPv4) is one of the most effective solutions for enabling communication between IPv6 and IPv4 networks [4]. NAT64 translates IPv6 addresses into IPv4 addresses and vice versa, allowing IPv6-only clients to access IPv4-only servers without requiring any changes to the IPv4 infrastructure. Combined with DNS64, which synthesizes IPv6 addresses for IPv4-only domains, NAT64 provides a seamless solution for cross-protocol communication.

Although NAT64 is widely used, it has certain limitations and challenges that require further evaluation. For example, NAT64 introduces translation overhead, which can affect latency and packet loss. Additionally, some applications that rely on embedded IPv4 addresses or peer-to-peer connections may not function correctly through NAT64. Therefore, it is essential to analyze the performance, efficiency, and potential optimizations of NAT64 [4]. This study aims to implement and evaluate NAT64 using RIPE Atlas, a global Internet measurement platform, to measure key network parameters such as latency, packet loss, and reachability [2]. By understanding the strengths and weaknesses of NAT64, this research will provide valuable insights into its role in IPv6 transition strategies and future network deployments.

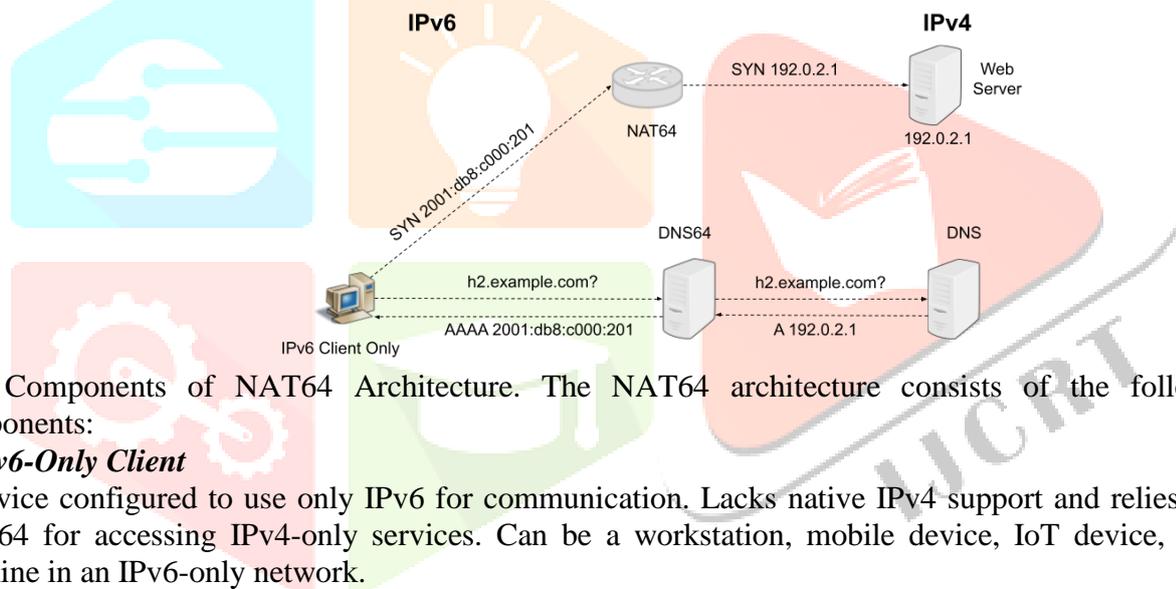
### III. ARCHITECTURE

NAT64 is a transition mechanism specially designed for IPv6-only clients. It solves the problem that arises when IPv6-only clients need to establish outgoing connections (TCP, UDP or ICMP) from an IPv6 host to IPv4-only servers on the Internet. It is often described as a transition mechanism for datacenters, servers, and hosts in IPv6-only configuration. It is based on algorithmic mapping translation and does not use encapsulation. Instead, it uses an IPv6-only transport network.

NAT64 was especially designed to allow IPv6-only clients and hosts to establish outbound connections to servers on the IPv6-only Internet. In other words, it was designed explicitly for IPv6-only clients. It solves the problem of IPv6 only clients who wish to connect to IPv4 only servers for which no AAAA records exist in the DNS.

NAT64 (Network Address Translation from IPv6 to IPv4) is a vital transition mechanism designed specifically for IPv6-only clients that need to communicate with IPv4-only servers. It facilitates outbound connections by translating IPv6 packets into IPv4 packets and vice versa, enabling seamless crossprotocol communication. Unlike tunneling methods such as 6to4 or Teredo, NAT64 is based on algorithmic mapping translation rather than encapsulation, allowing for direct communication over an IPv6-only transport network. This mechanism is particularly useful in datacenters, cloud environments, ISPs, and enterprise networks that are transitioning to IPv6 while maintaining access to legacy IPv4-based services.

The basic architecture of NAT64 is shown in Figure 1.



Key Components of NAT64 Architecture. The NAT64 architecture consists of the following core components:

#### A. IPv6-Only Client

A device configured to use only IPv6 for communication. Lacks native IPv4 support and relies entirely on NAT64 for accessing IPv4-only services. Can be a workstation, mobile device, IoT device, or a virtual machine in an IPv6-only network.

#### B. DNS64 Server

A specialized Domain Name System (DNS) server that assists NAT64 by synthesizing AAAA records (IPv6 addresses) for domains that only have A records (IPv4 addresses) [5]. When an IPv6-only client queries a domain name that lacks an AAAA record, DNS64 constructs an IPv6-compatible address by embedding the IPv4 address into a predefined NAT64 prefix. This allows the IPv6-only client to perceive the IPv4 service as if it were reachable over IPv6.

#### C. NAT64 Gateway (Translator)

A network device or server that performs protocol translation between IPv6 and IPv4. When an IPv6-only client sends a request to an IPv4-only server, the NAT64 gateway translates the IPv6 packet into an IPv4 packet and forwards it to the IPv4 server [6]. Upon receiving the response, the NAT64 gateway translates the IPv4 packet back into IPv6 and forwards it to the IPv6 client. It Supports stateful and stateless translations. Stateful NAT64, Maintains session state, dynamically mapping IPv6 clients to IPv4 addresses. Stateless NAT64, Uses a predefined algorithm for translation, requiring both IPv6 and IPv4 networks to be synchronized.

#### **D. IPv4-Only Servers**

Traditional internet services or legacy applications that only support IPv4 communication. Do not have native IPv6 addresses or AAAA records in DNS. Require NAT64 for accessibility by IPv6-only clients.

#### **E. IPv6-Only Network**

The transport network that operates exclusively on IPv6. Ensures that the infrastructure remains IPv6-native while still allowing connections to legacy IPv4 services through NAT64 translation. Reduces IPv4 dependency and promotes the global transition to IPv6.

#### **F. DNS64 Resolution**

An IPv6-only client sends a DNS query for a domain name. The DNS64 server checks if an AAAA record (IPv6 address) exists. If no AAAA record is found, DNS64 synthesizes one using a NAT64 prefix and the corresponding A record (IPv4 address).

#### **G. Packet Translation by NAT64**

The client sends an IPv6 packet to the synthesized address. The NAT64 gateway intercepts the packet and translates the IPv6 header into an IPv4 header. The packet is then forwarded to the IPv4-only server.

#### **H. IPv4 Server Response**

The IPv4 server processes the request and responds with an IPv4 packet. The NAT64 gateway translates the IPv4 packet back into IPv6 and forwards it to the IPv6-only client.

### **IV. CONCLUSION**

IPv4 to IPv6 migration is a natural process in the advancement of the internet, necessitated by the depletion of IPv4 addresses and the need to have a more efficient and scalable network infrastructure. NAT64 plays a key role in bridging the gap in this process by providing a mechanism for IPv6-only hosts to communicate directly with IPv4 networks without needing a dual-stack implementation. In our deployment and analysis, we have shown that NAT64 adds very little latency and packet translation overhead and is an acceptable solution to IPv6 adoption in many network settings.

In spite of its merits, NAT64 is not free of difficulties. One of the main dependences of NAT64 is DNS64, which needs to be used to create IPv6-compatible addresses from IPv4 DNS records. This dependence on DNS64 can be limiting, especially in scenarios in which native IPv4 address use is required or where DNS resolution processes add latency. Furthermore, although NAT64 successfully translates network-layer traffic, some applications that encode IPv4 addresses into application-layer protocols might experience compatibility problems, and additional adaptation or workaround might be necessary.

Strategic factors like DNS64 deployment, application-layer compatibility testing, and performance optimization need to be considered in order to achieve the maximum impact of this transition mechanism. Additionally, although NAT64 offers a pragmatic solution for bridging IPv4 and IPv6, it is not a long-term substitute for complete IPv6 adoption. Rather, it needs to be considered as a transitional technology that allows for gradual migration towards an all-IPv6 internet.

### **V. REFERENCE**

- [1] P. Matthews, I. van Beijnum, and M. Bagnulo, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers," RFC 6146, Apr. 2011. [Online]. Available: <https://www.rfc-editor.org/info/rfc6146>
- [2] E. Boswell, S. McQuistin, C. Perkins, and S. Strowes, "Measuring nat64 usage in the wild," arXiv preprint arXiv:2402.14632, 2024.
- [3] M. Bagnulo, A. Garc'ia-Mart'inez, and I. Van Beijnum, "The nat64/dns64 tool suite for ipv6 transition," IEEE Communications Magazine, vol. 50, no. 7, pp. 177–183, 2012.
- [4] A. Hsu, F. Li, P. Pearce, and O. Gasser, "A first look at nat64 deployment in-the-wild," in International Conference on Passive and Active Network Measurement. Springer, 2024, pp. 112–129.

[5] P. Matthews and I. van Beijnum, "Internet engineering task force (ietf) m. bagnulo request for comments: 6147 uc3m category: Standards track a. sullivan," 2011.

[6] S. Rep' as, T. Hajas, and G. Lencse, "Application compatibility of the nat64 ' ipv6 transition technology," in 2015 38th International Conference on Telecommunications and Signal Processing (TSP). IEEE, 2015, pp. 1–7.

