ASTES

Advances in Science, Technology and Engineering Systems Journal Vol. 6, No. 1, 336-341 (2021)

www.astesj.com

Special Issue on Multidisciplinary Sciences and Engineering

ASTESJ

ISSN: 2415-6698

Recent Impediments in Deploying IPv6

Ala Hamarsheh*,[1], Yazan Abdalaziz[1], Shadi Nashwan[2]

[1]*Computer Science, Arab American University, Jenin, 240, Palestine*

[2]*Computer Science Department, College of Computer and Information Sciences, Jouf University, Sakaka, 2014, Saudi Arabia*

ARTICLE INFO

Article history:
Received: 20 November, 2020
Accepted: 06 January, 2021
Online: 22 January, 2021

Keywords:
Internet Protocol Version 6
Tunneling
Translation
IPv6 Adoption
IPv6 transition techniques

ABSTRACT

Internet Protocol version 6 is being adopted on slow pace and it is taking a long time. This paper intends to discuss the transition process between IPv4 and IPv6 and the major obstacles that prevent deploying IPv6 worldwide. It presents the IPv4 exhaustion reports results and where are the IPv4 address pool. Then it presents the methods that have been used to prolong the life expectancy of IPv4. After that it describes and discusses the mechanisms that have been used to deploy IPv6. Additionally, it describes the recently proposed mechanisms to overcome the problems encountered by the ISPs in migrating to IPv6. Furthermore, it shows the mechanisms that have been proposed to motivate the ISPs to start deploying IPv6 on their access networks. Finally, it presents a comparison between these mechanisms from the authors' point of view.

1. Introduction

'Internet Protocol Version 4" (IPv4) [1] is the current considered primary Internet protocol. It uses a 32-bit address. That provides more than 4 Billion addresses. Due to the frequent use of the internet and the abundance of devices that need an Internet connection, IPv4 was drained. As a temporary solution, many mechanisms have been proposed to prolong the life of IPv4. In the end, there must be a permanent solution. Internet Engineering Task Force (IETF) proposed a solution in which to convert IPv4 to a new protocol which is the "Internet Protocol Version 6" (IPv6) [2].

IPv6 was created by the IETF as a successor protocol to IPv4. It was proposed in 1994 and it became an internet standard on 14 July 2017 [3]. IPv6 uses a 128-bit address which provides approximately 3.4 x 1028. That will fulfill the huge need for IP addresses. The problem is that IPv6 can't be deployed all at once because of the large number of users and devices. Also, the lack of readiness of the infrastructure may delay the process knowing that the users can't be forced to update or change their devices so that they can use IPv6. IPv4 and IPv6 are not compatible and they can't be deployed both at the same time. IETF has proposed many solutions to start deploying IPv6 alongside IPv4. The plan is to start integrating IPv4 with IPv6 until IPv6 is fully deployed.

This Paper is an extension of work originally presented in [4]. In this research paper, we will present the methods that have been used to conserve the public IPv4 addresses. Then we will present some reasons that prevent the IPv6 deployment process. After that, we will present the mechanisms that have been proposed to integrate IPv6 alongside IPv4. Finally, we will compare and discuss these mechanisms from the authors' point of view.

2. IPv4 Exhaustion

The "Internet services providers" ISPs get the IP addresses from the Regional Internet Registry (RIR). There are five RIRs each one is responsible for providing IP addresses in a particular area of the world. which are: "American Registry for Internet Numbers (ARIN), Asia-Pacific Network Information Center (APNIC), Latin America and Caribbean Information Center (LACNIC), Reseaux IP Europeens Network Coordination Center (RIPE NCC), and the African Network Information Center (AFRINIC)". A statement has been made by APNIC that they have been operating under the last/8 framework since April 2011. At that time, they declared that they have 11,028,480 addresses assigned as available and have 15,728,128 unassigned addresses. For more information about the other RIRs reports check [5].

3. Methods to Preserve "Public IPv4 Addresses"

Many mechanisms that have been proposed to prolong the life expectancy of IPv4 the first one is the "Network Address Translation (NAT)" [6]. In short, NAT is a translation mechanism that translates IPv4 public addresses to a group of IPv4 private

*Corresponding Author: Ala Hamarsheh, Jenin, P.O Box 240, Palestine, E-mail: ala.hamarsheh@aaup.edu

https://dx.doi.org/10.25046/aj060138

addresses. rapid growth of internet users and IP addresses consumption has affected on NAT and it is not useful solution anymore. However, IETF has proposed some mechanisms which are:" Large Scale NAT (NAT444)" [7], and "Address plus Port (A+P)".

3.1. Large Scale NAT (NAT444)

Large Scale NAT or NAT444 is a technology that has been considered by the ISPs to extend the life of IPv4. Using NAT444 ISPs have been able to provide the users with IPv4 during the transition to the IPv6 process. It works by adding a Carrier Grade NAT (CGN) to the ISP's network. It translates one public IPv4 address among various CPEs and the private address in the CPE will again be translated into many private IPv4 addresses to arrive at the end-user with two NATs. NAT444 has many disadvantages, for instance, users may affect each other on bad behaviors because they share one public IPv4 address among a large number of users. Also, losing geolocation information because translation zones will cross traditional geographic boundaries see [8]. VoIP and video applications might be affected by NAT444 with latency, or packet loss, etc. see [9].

3.2. Address Plus Port (A+P)

A+P is a sharing schema that allows multiple users on the same CPE to use the same public IPv4 address. It extends the IPv4 address by using some of the port numbers as additional endpoints identifiers in the TCP/UDP header. Port Range Router (PRR) runs the process of assigning the range of IP addresses among the CPEs. A+P schema can provide 65536 ports for each public IPv4 address and each port can provide an extra 65536 ports. The idea behind A+P is that it divides the public IPv4 address without the need for translating it see [10] for more details.

4. Transition Approaches

Since IPv6 and IPv4 are not corresponding to each other and deploying IPv6 cannot be done all at once, the IETF has proposed techniques and mechanisms in order to achieve a smooth transition to IPv6. The smooth transition can be achieved by deploying IPv6 at the same time as IPv4. To allow communications between IPv4-Only networks and IPv6-Only networks IETF has proposed the transition mechanisms. These mechanisms are Dual-Stack, Tunneling, and Translation Approaches.

4.1. Dual-Stack Approach

The dual-stack approach is providing the user's connectivity for both IPv4 networks and IPv6 networks. A dual-stack device is any device that connects to the internet and configured with multiple IP addresses IPv4 and IPv6 at the same time. The idea behind the dual-stack approach is to configure all devices with both IPv4 and IPv6 addresses so that the devices can communicate over both networks [11]. It uses the perspective protocols (DHCPv4, DHCPv6) [12], [13] to assign addresses to the dual-stack devices. The network administrator is responsible for enabling perspective protocols. See Figure 1 which describes the Dual-Stack approach.

4.2. Tunneling Approach

The tunneling approach is a protocol that allows the movement of data from a network to another through a different type of network. IPv4/IPv6 Routers and Hosts encapsulate the IPv6 packet in an IPv4 packet and send it through an IPv4 network to another IPv6 network. Tunneling can be used in various ways for instance Router-to-Router, Host-to-Router, Host-to-Host, and Router-to-Host. Because the hosts and routers need to explicitly configure the tunneling endpoints Router-to-Router is probably used. Figure 2 illustrates the Tunneling approach.

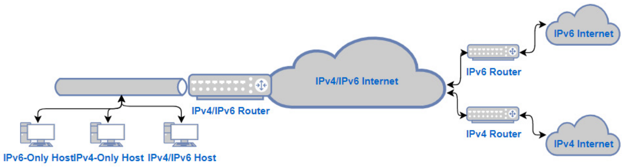

Figure 1: Dual-Stack Approach

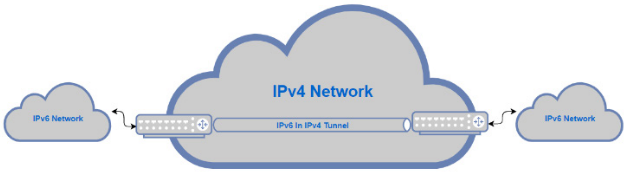

Figure 2: Tunneling Approach

Tunneling has two types which are: Static Tunneling and Automatic Tunneling.

- Static Tunnels: in static tunneling the address will be configured at the tunnel endpoint and the configuration in this technique is manual. The encapsulating node is the one responsible for storing the tunneling information. It stores the tunnel endpoint address which will be the destination address. in addition, it stores the routing information in order to determine which packet to tunnel. Also, it uses the match technique and the prefix mask to the destination to the packet [14]-[16].
- Automatic Tunnels: in automatic tunneling the IPv6/IPv4 nodes can automatically determine the tunnel endpoint and That can be extracted from the IPv6 address. definitely the IPv6 must be backward compatible with IPv4. Also, the nodes possess the ability to decide which packets are auto tunneled and which are not. The main difference between the dynamic and static tunneling is the automatic determination of the tunnel endpoint.

4.3. Translation Approach

The previous approaches could be useful when communication is needed between two isolated IPv4 networks or two isolated IPv6 networks, while they have no use when an IPv4 only network tries to communicate with an IPv6 only network. The translation approach is a mechanism that allows an IPvX-Only network to communicate directly with an IPvY-only network. It translates the IPvX packet to an IPvY packet to allow communication. Figure 3 describes the translation approach.

The translation mechanism can be classified into two approaches [11]: Host-Based Approach and Network-Based Approach.

- Host-Based Translation

  Host-based translation is needed when there is an incompatibility between the running application type and the current host connectivity. So that, the IPvX will be translated to communicate with the IPvY and vice versa. In host-based translators, the changeover is between the application layer and the IP communication layers. There are three host-based translators was proposed which are: Bump-In-the-Stack (BIS) [17], Bump-In-the-API (BIA) [18], Bump-In-the-Host (BIH) [19]. In the next section we will present the host-based techniques in addition to another technique the author was proposed which is Decupling Application IPv4/IPv6 Operation from the Underlying IPv4/IPv6 Communication (DAC) [20].

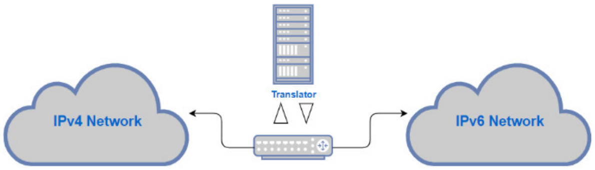

Figure 3: Translation Approach

- Network-Base Translation

  In network-based translation, the IP header will be translated to every packet. That to provide the connectivity between IPv4-only networks and IPv6-only networks and vice versa [11]. An example of network-based translators is Stateless IP/ICMP Translation (SIIT) [21], and Stateful Address and Protocol Translation from IPv6 Client to IPv4 Servers (NAT64) [22].

5. IPv6 is Not Fully Deployed Yet

Transition to IPv6 is indeed a big breaking change in the networks. It can’t be done all at once for many reasons. This section intends to describe these reasons from the author's point of view.

5.1. End User

End-users are not directly benefiting from the transition process; however, it is considered as an effective part in achieving it. The end-user might pose an obstacle for deploying IPv6. As we discussed earlier the deployment process of IPv6 must be smooth. A smooth transition can be done without affecting the users taking into account the users’ application and the host configurations should be upgraded. The end-user can delay the process knowing that not all users have the proper knowledge to update or implement the upgrades.

There are some upgrades that should be implemented on the IP stack, “Transmission Control Protocol (TCP)”, and “User Datagram Protocol (UDP)” in addition to the users’ old applications [11], [23], [24]. The host configuration process should be automatically performed without the user knowledge because the user can’t be expected to be qualified to deal with the manual configurations or any other technical configurations.

Achieving the transition process smoothly could take a while till IPv6 is fully deployed. Within this time there will be IPv6 only networks and IPv4 only networks that need to be able to communicate with each other. The solution to that problem is the dual connectivity so that dual networks can communicate to both IPv4 and IPv6 networks [25]. The users’ applications can be expected to be updated to support dual connectivity if it has good support but there are many applications that are not supported such as home-made applications and out-of-business applications and many other applications that will have the same problem.

There are many mechanisms that have been proposed as a solution to guarantee dual connectivity between incompatible hosts. IETF has proposed mechanisms such as “Bump In The Stack (BIS)” [17], “Bump In the API (BIA)” [18], [26], and “Bump In The Host (BIH)” [19]. Another mechanism that has been proposed by the author of this paper is “Decupling Application IPv4/IPv6 operation from the Underlying IPv4/IPv6 Communication (DAC)” [20], [27], [28].

- Bump in the Stack (BIS)

Bump-In-the-Stack is a technique that uses the SIIT [29] algorithm to translate the IPv4 packets (i.e. headers) to IPv6 and the IPv6 packets (i.e. headers) to IPv4. In BIS the translation module is inserted between the TCP/IPv4 module and the card driver module. The translation module takes the data that flow between the modules that it is inserted between and translates the packets from IPv4 to IPv6 and vice versa. The DNS server is responsible for assigning the IP addresses. So, the user does not know about the other types of IP addresses that it communicates with. The main idea behind BIS is to provide the communication between IPv4 applications and IPv6 hosts.

- Bump in the API (BIA)

Is a technique that used to translate the IPv4 socket API functions into IPv6 socket API functions and vice-versa. It detects the IPv4 socket API functions and invokes the equivalent IPv6 socket API functions so that there will be no need for translating the full IP header. BIA expects that TCP/UDP IPv4 and TCP/UDP IPv6 to existing on the local node. Unlike BIS, BIA inserts the API translator between the socket API module and the TCP/IP module in the dual-stack hosts.

- Bump in the Host (BIH)

BIH technique is a host-based mechanism. It integrates both BIS and BIA techniques together and translates from IPv4 to IPv6. BIH consists of two implementations which are a translator in the API socket which is inserted between the TCP/IP module and socket IP module the other implementation is the translation protocol which inserts it between the network card driver and the TCP/IP module. BIS intends to let the old applications which uses NAT to communicate with the IPv6 only applications.

- Decoupling Application IPv4/IPv6 Operation from the Underlying IPv4/IPv6 Communication (DAC)

DAC is a technique Allows applications that are compatible with IPv4-only stack and running on hosts with dual stack or with IPv6-only connectivity to communicate with IPv6 hosts. Additionally, it enables applications that are compatible with IPv6-only stack and running on hosts with dual stack and connectivity or with IPv4-only connectivity to communicate with IPv4 hosts. In theory, DAC is a layer that should be inserted between the application layer and the IP communication stack on top of the API functions and it should perform the same functionalities that the API native functions perform. Translation from IPv4 and IPv6 and vice versa is only done on necessary basis. DAC will be effective between applications with an incompatible type and an incompatible networks connectivity. Furthermore, DAC can be installed at an IPv4 only host, IPv6 only host, and IPv4/IPv6 host.

5.2. Service Provider Network

The Internet Service Providers (ISPs) are the other effective part of the deployment process. as we discussed before the ISPs are cannot provide the users IPv4 addresses anymore and they need to deploy the IPv6 so that they can keep up with the internet users' swift increasing and provide the users with the required IP addresses. The transition process requires an upgrade or change of the infrastructure to one that can withstand the IPv6. That might be very expensive for the ISPs. IETF has proposed a technique that allows the ISPs to start deploying IPv6 over IPv4 networks. This technique is IPv6 Rapid Deployment on IPv4 Infrastructure (6rd) [30], [31].

6rd is an automatic tunneling technique was proposed to support rapidly deployment of IPv6 on existing IPv4 environment. In order to transmit the IPv6 traffic over IPv4 environment, 6rd uses encapsulation technique to transfer IPv6 packets through the IPv4 networks and it uses the ISPs’ own IPv6 prefix other than a well-known prefix. Since it uses the network-specific prefix (NSP) each ISP can use its particular prefix that’s mean the 6rd operational scope is limited to the ISP’s domain. Moreover, the tunnels will be from the ISP’s border relay (6rd gateway) to the customer edge (CE). see [30] and [32] for more details.

ISPs have faced determinations that hindered them from starting the deployment process of IPv6 through IPv4 infrastructure. One of these limitations is the need to upgrade or change the 6rd gateways (CE) which will affect them by raising the costs of deploying IPv6. In addition to the time that the process of Configuring the CEs would take after change and upgrade which might stall the process. Also, 6rd protocol upholds only one level of DHCPv4 between the CE and the border router. Never forget to mention the firewalls which might block the tunneled traffic that related to 6rd.

The authors of this research paper have suggested a few techniques that concurs the 6rd obstacles. These techniques are: “Deploying IPv6 Service Across Local IPv4 Access Network (D6across4)” [33], [34], “Configuring hosts to Auto-detect (IPv6, IPv6-in-IPv4, or IPv4) network connectivity (CHANC)” [35], and “Deploying IPv4-only Connectivity across Local IPv6-only Access Networks (D4across6)” [36]. The next section of this research will discuss these techniques and how they have overcome the limitations and obstacles that faced the ISPs before.

- Deploying IPv6 Service Across Local IPv4 Access Network (D6across4)

D6across4 is a mechanism that intends to motivate the ISPs to start the deployment process and start offering IPv6 services to their customers. It is a protocol mechanism to start deploying IPv6 service to host over the existing ISP’s IPv4 network. D6across4 assigns an algorithmic mapping between the IPv4 and IPv6 addresses in the ISP network. It automatically resolves the tunnel server (TS) domain name to determine the tunnel endpoints. It consists of customers’ hosts and one or more TSs. Same as in 6to4 it encapsulates the IPv4 packets then forwards them to follow the topology of the IPv4 in the ISP network.

While 6rd and other protocols need upgrade and change the 6rd gateways and the network devices in both ISPs’ and end users’ sides. D6across4 tries to reduce both cost and time of the deployment process of IPv6 which is one of the problems that faced the ISPs and that prevented them from starting. Even though, in order for D6across4 can encapsulate/de-capsulate traffic IPv6 packets through the local IPv4 networks, it needs to be installed at ISPs’ side at the network components, and at the uses side particularly at the applications hosts. See for more details [33].

There are some limitations that could be found in D6across4 that could affect deploying IPv6. These limitations are: if D6across4 has been deployed on a large scall networks it could face some issues regarding the scalability and the performance issues. So, it only could be integrated on a relatively small network. Also, since the D6across4 protocol is a stateful operation, for each packet receival it needs to access the mapping tables that could be exhaustion and consider as an additional load and it would be higher than any other stateless protocol. In addition, using a well-known prefix might shorten IPSs’ capability of managing and controlling their traffic. Just like 6rd, D6across4 could be affected by firewalls and its traffic might be blocked.

- Configuring hosts to Auto-detect (IPv6, IPv6-in-IPv4, or IPv4) network connectivity (CHANC)

Same as D6across4 CHANC is a mechanism to encourage the ISPs to start the deploying IPv6 through IPv4 environment. Unlike D6across4 CHANC protocol doesn’t need any upgrading or changing in the CEs which represents the extra costs that cause the delay of the deployment process. CHANC protocol gives the ISPs the ability to offer IPv6 to their customers through IPv4 infrastructure with the minimum cost and automatically configure the end-user’s host. In fact, CHANC automatically configure the hosts to give them the ability to automatically detect every connectivity with every type that their ISP provide and locating the relay server would also be automatic. Also, CHANC uses the HTTP-in-IPv4 mechanism for transmitting IPv6 traffic through IPv4 environment.

CHANC has many advantages that overcome the other proposed protocols which are:

- The configurations in the end-users’ hosts are done automatically which ease the process instead of manual configurations.
- It reduces the costs because there is no need for changing or upgrading the IPv4-only networks on both ISPs’ and end-users’ sides.
- Because of the IPv4 only access network, the ISPs can immediately begin offering IPv6 without the need support IPv6.
- It prevents the firewalls from blocking the tunneled packets of CHANC. It encapsulates the packets in HTTP protocol which permit the access to the traffic through utmost all firewalls.
- It simplifies administrating IPv6 traffic by allowing every ISP to use the prefix of its own.

Table 1: Comparison between the previously presented techniques and mechanisms

Protocol	**Category**	**Installed At**	**Functionality**	**Limitations**
BIS	Translation-based	"between the TCP/IP module and the network card driver"	Permit communication between IPv4 only applications on dual stack machines and IPv6 hosts	- Does not work with multicast communication. - invalid for embedded addresses. - It cannot be combined with a secure DNS. - It cannot employ security overhead the network layer.
BIA	Translation-based	"between the socket API module and the TCP/IP module"	Permit communication between IPv4 only applications on dual stack machines and IPv6 hosts	- invalid embedded addresses. - Does not uphold multicast. - Difficulties in translating APIs because of IPv6 API's advance new features.
BIH	Translation-based	"between the TCP/IP module and the network card driver or between socket API module and the TCP/IP module"	Permit communications between IPv4 legacy application and IPv6 only hosts and dual stack host	- invalid for embedded addresses. - Does not uphold multicast. - Does not uphold all types of applications.
DAC	Translation-based	"between the communication stack and the application layer"	Permit communications for both IPv4 and IPv6 applications through IPv4 and IPv6 connections with both IPv4 and IPv6 type capable remote applications.	- invalid for embedded addresses.
6rd	Tunneling-based	between the ISP border relay and the customer edge	Permit IPv6 connectivity through ISPs' IPv4 environment	- Single point of failure. - requires upgrade and change in the CPEs. - Cannot cross firewalls.
D6across4	Tunneling-based	ISP side and end users' host	Permit IPv6 connectivity through ISPs' IPv4 environment	- Single point of failure. - cannot cross firewalls.
CHANC	Tunneling-based	ISP side and end users' host	Permit IPv6 connectivity through ISPs' IPv4 environment	- Single point of failure.

- CHANC is a stateless-based translation mechanism that uphold end to end address translucence while the communication is between IPv4 and IPv6.

The Table below presents a comparison between the techniques and protocols that have been presented in this research.

6. Summary

The internet development has led to exhaustion in the IPv4 addresses and most of the ISPs are running out of IPv4. That makes the deployment of IPv6 has become mandatory. IETF has offered various mechanisms to extend the usage of IPv4. The proposed mechanisms and protocols are considered as a temporary solution. Afterall, the deployment of the IPv6 is unavoidable. ISPs are the major factor that affect the deployment directly. They need to start upgrading and changing the infrastructure to support IPv6. Also, IETF has offered various techniques and protocols to encourage the ISPs to start offering IPv6 alongside with IPv4 or even IPv6 through IPv4 infrastructure. These techniques and protocols can be classified into three classifications: Dual stack, Tunneling, and translation. In addition, various techniques and mechanisms have been offered by the authors of this research and by IETF for instance BIS, BIA, BIH, DAC, 6rd, D6across4, and CHANC. These mechanisms intend to encourage the ISPs and facilitate the deployment process in order for them to start the smooth transition.

Conflict of Interest

Ala Hamarsheh, Yazan Abdalaziz, Shadi Nashwan declare that they have no conflict of interest.

References

[1] J. Postel, "Internet Protocol," California, USA: Internet Engineering Task Force RFC791, 1981.

[2] S. Deering, R., Hinden, "IP Version 6 Addressing Architecture," California, USA: Internet Engineering Task Force RFC 4291, 2006.

[3] S. Deering, R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," California, USA: Internet Engineering Task Force RFC 8200, 2017.

[4] A. Hamarsheh, Y. Abdalaziz, "Deploying IPv6 Service Across Local IPv4 Access Networks," in 2019 International Conference on Computer and Information Sciences (ICCIS), Sakaka, Saudi Arabia, April 3-4, 2019.

[5] G. Huston. IPv4 Address Report [Online]. 2018. Available: http://www.potaroo.net/tools/ipv4/index.html

[6] P. Srisuresh, M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations," California, USA: Internet Engineering Task Force RFC 2663, 1999.

[7] I. Yamagata, Y. Shirasaki, A. Nakagawa, J. Yamaguchi, H. Ashida, "NAT444," Internet Engineering Task Force Draft (work in progress), 2012.

[8] C. Donley, L. Howard, V. Kuarsingh, J. Berg, J. Doshi, "Assessing the Impact of Carrier-Grade NAT on Network Applications," California, USA: Internet Engineering Task Force RFC 7021, 2013.

[9] S. Bradner, J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices," California, USA: Internet Engineering Task Force RFC 2544, 1999.

[10] R. Bush, "The Address plus Port (A+P) Approach to the IPv4 Address Shortage," California, USA: Internet Engineering Task Force RFC 6346, 2011.

[11] A. Hamarsheh, M. Goossens, "A Review: Breaking the Deadlocks for

Transition to IPv6," IETE Technical Review, **31**(6), 405-421, 2014, doi: 10.1080/02564602.2014.950348

[12] E. Nordmark, R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers," California, USA: Internet Engineering Task Force RFC 4213, 2015.

[13] A. Durand, R. Droms, J. Woodyatt, Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion," California, USA: Internet Engineering Task Force RFC 6333, 2011.

[14] R. Gilligan, E. Nordmark, "Transition Mechanisms for IPv6 Hosts and Routers," California, USA: Internet Engineering Task Force RFC, 1996.

[15] Y. Abdalaziz A. Hamarsheh, "Analyzing The Ipv6 Deployment Process In Palestine", International Journal of Computer Network and Information Security(IJCNIS), **12**(5),31-45, 2020, DOI: 10.5815/ijcnis.2020.05.03.

[16] M. Al-Fayoumi S. Nashwan, "Performance analysis of SAP-NFC protocol," International Journal of Communication Networks and Information Security (IJCNIS), **10**(1), 125–130, 2018.

[17] K. Tsuchiya, H. Higuchi, Y. Atarashi, "Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS)," California, USA: Internet Engineering Task Force RFC 2767, 2000.

[18] S. Lee, M-K. Shin, Y-J. Kim, E. Nordmark, A. Durand, "Dual Stack Hosts Using "Bump-in-the-API" (BIA)," California, USA: Internet Engineering Task Force RFC 3338, 2002.

[19] B. Huang, H. Deng, T. Savolainen, "Dual-Stack Hosts Using "Bump-in-the-Host" (BIH)," California, USA: Internet Engineering Task Force RFC 6535, 2012.

[20] A. Hamarsheh, M. Goossens, R. Alasem, "Decoupling Application IPv4/IPv6 Operation from the Underlying IPv4/IPv6 Communication (DAC)," American Journal of Scientific Research, Eurojournals Press, **14**, 101-121, 2011.

[21] X. Li, C. Bao, F. Baker, "IP/ICMP Translation Algorithm," California, USA: Internet Engineering Task Force RFC 6145, 2011.

[22] M. Bagnulo, P. Matthews, I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers," California, USA: Internet Engineering Task Force RFC 6146, 2011.

[23] Y. Abdalaziz, A. Hamarsheh, "Analyzing The IPv6 Deployment Process In Palestine," International Journal of Computer Network and Information Security(IJCNIS), **12**(5), 31-44, 2020, doi: 10.5815/ijcnis.2020.05.03

[24] S. Nashwan, "SAK-AKA: A secure anonymity key of authentication and key agreement protocol for LTE network," International Arab Journal of Information Technology (IAJIT), **14**(5), 790–801, 2017.

[25] A. Hamarsheh, M. Goossens, A. Al-Qerem, "Assuring Interoperability Between Heterogeneous (IPv4/IPv6) Networks Without using Protocol Translation," IETE Technical Review, **29**(2), 114-132, 2012, doi:10.4103/0256-4602.95384

[26] S. Nashwan, "SE-H: Secure and efficient hash protocol for RFID system," International Journal of Communication Networks and Information Security (IJCNIS), **9**(3), 358–366, 2017.

[27] A. Hamarsheh, M. Eleyat, "Performance Analysis Of Ain-Pt, Ain-Slt And Siit Network-Based Translators," in Advances on P2P, Parallel, Grid, Cloud and Internet Computing Proceedings of the 12th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC-2017), Lecture Notes on Data Engineering and Communications T. Springer, 10, Palau Macaya, Barcelona, Spain, 367-378, 2017, DOI: https://doi.org/10.1007/978-3-319-69835-9_35.

[28] S. Nashwan, "AAA-WSN: Anonymous access authentication scheme for wireless sensor networks in big data environment," Egyptian Informatics Journal, in press, https://doi.org/10.1016/j.eij.2020.02.005.

[29] Nordmark E, "Stateless IP/ICMP Translation Algorithm (SIIT)," Internet Engineering Task Force RFC 2765, 2000.

[30] W. Townsley, O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) Protocol Specification," California, USA: Internet Engineering Task Force RFC 5969, 2010.

[31] A. Al-Qerem, F. Kharbat, S. Nashwan, S. Ashraf and K. Blaou, "General model for best feature extraction of EEG using discrete wavelet transform wavelet family and differential evolution," International Journal of Distributed Sensor Networks, **16**(3), 1–21, 2020.

[32] R. Despres, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd)," California, USA: Internet Engineering Task Force RFC 5569, 2010.

[33] A. Hamarsheh, M. Goossens, R. Alasem, "Deploying IPv6 Service Across Local IPv4 Access Networks," in 10th WSEAS International Conference on TELECOMMUNICATIONS and INFORMATICS (TELE-INFO '11), pp. 94-100, Lanzarote, Canary Islands, Spain, 27-29, 2011.

[34] A. Hamarsheh, M. Goossens, "Exploiting Local IPv4-only Access Networks to Deliver IPv6 Service to End-users," International Journal of Computers and Communications, **5**(3), 2011.

[35] A. Hamarsheh, M. Goossens, R. Alasem, "Configuring Hosts to Autodetect (IPv6, IPv6-in-IPv4, or IPv4) Network Connectivity," KSII Transactions on Internet and Information Systems," **5**(7), 1230-1251, 2011, doi:10.3837/tiis.2011.07.002.

[36] A. Hamarsheh, "Deploying IPv4-only Connectivity across Local IPv6-only Access Networks," IETE Technical Review, **36**(4), 398-411, 2018, doi: 10.1080/02564602.2018.1498031.