Wireless Pers Commun (2016) 87:381–398
DOI 10.1007/s11277-015-2876-2

Green Computing in WAN Through Intensified Teredo IPv6 Tunneling to Route Multifarious Symmetric NAT

Sheryl Radley[1] · **D. Shalini Punithavathani**[1]

Published online: 15 July 2015

Abstract IPv4–IPv6 transition rolls out numerous challenges to the world of Internet as the Internet is drifting from IPv4 to IPv6. IETF recommends few transition techniques which includes Dual stack, translation and tunneling. By means of tunneling the IPv6 packets over IPv4 UDP, Teredo maintains IPv4/IPv6 Dual stack node in isolated IPv4 networks behindhand Network Address Translation (NAT). However, the proposed tunneling protocol works with the symmetric and asymmetric NAT. In order to make a Teredo support several symmetric NATs along with several asymmetric NAT, we propose Multifarious Sym Teredo, which is an extension of Teredo with a capability of navigating through several symmetric NAT using Graphical Network Simulator 3. The work preserves the Teredo architecture and also offers a backward compatibility with the original Teredo protocol.

Keywords IPv4 · IPv6 · Tunneling · Teredo · NAT · Symmetric NAT · Asymmetric NAT

1 Introduction

IPv4, the first version of the internet protocol offers a unique global computer addressing [1] to make sure two entities can exceptionally identify one another. Due to the evolution in the number of users day to day, IPv4 has lost its pace. The next generation IP (IPng), IPv6 has been designated from several proposed alternatives as a suitable replacement of the existing protocol, since it provides sufficient IP addresses to enable all categories of devices to connect to the internet. The IETF Next Generation Transition Working Group (NGtrans) has proposed many transition mechanisms to enable the unified integration of IPv6 facilities into existing networks [2]. The transition mechanisms are proposed to create

✉ Sheryl Radley
phdsheryl@gmail.com

[1] Department of CSE, Government College of Engineering, Tirunelveli, Tamil Nadu, India

a smooth transition [3]. Deployment of Internet Protocol version 6 (IPv6) in the internet has been comparatively slow since its introduction over a decade ago. There are variety of business and practical reasons for the low prevalence of the IPv6 networks. The reason behindhand this is the backbone of the network cannot be changed overnight. Number of techniques has been proposed over the ages to upkeep the continuous growth of the global Internet required for overall architecture development to accommodate the new technologies that support the increasing number of users. Applications, appliances and services such as NAT-PT, Bump in stack (BIS), stateless Internet Protocol Internet Control Messaging Protocol (SIIT), Static tunneling, Tunnel Broker, ISATAP, 6to4 tunneling, 6in4 tunneling, 6over4 tunneling, Teredo, NAT64 and 6rd (IPv6 rapid development) has been developed to support the interoperability between IPv4 and IPv6. IPv4–IPv6 transition and coexistence is only possible with techniques like Dual stack [4], translation and tunneling [5]. All the transition mechanism are considered as a set of method to facilitate a smooth transition to a new IP version, unfortunately not all of them are amenable to the users option [6]. Tunneling is one of the methods used to connect isolated IPv6 nodes through the IPv4 network which in turn facilitates the transition of IPv4 to IPv6 network migration. Traditional tunneling approaches transport IPv6 packets as payload of IPv4 packets. Nevertheless, these approaches fail when one endpoint of the tunnel is positioned in a private IPv4 network behind one or several Network Address Translation (NAT) [7] [8]. Umpteen IPv6 tunneling solutions have been proposed to resolve the NAT traversal concern. Among these solutions, IETF v6ops working group choose Teredo as the protocol for client to traverse NAT and automatically establish IPv6 tunnels in an unmanaged network [9]. A foremost advantage of Teredo is that its load balancing design that utilizes a centralized Teredo server for signaling and Teredo relay intended for data packets delivery [10] [11].

2 Tunneling Techniques

Tunneling is a system that allows IPv6 packets to be transmitted over an IPv4 network and vice versa. Tunneling can take place between two routers, two hosts or between a router and a host. The 6to4 mechanism functions by having the IPv4 address of the router's IPv4 interface are a portion of the prefix of the IPv6 addresses allotted to the IPv6 host in the corresponding IPv6 domain. When a tunnel is configured manually, it is quite possible that a tunnel do not always take an optimal path amongst sites, where one IPv6 hop may span many Ipv4 hops [12]. Automatic tunneling originates in the 6to4/4to6 edge router and IPv6/IPv4 is the subnet technology. Due to encryption and decryption, the CPU utilization is high. Fragmentation issue also arises. Different tunneling techniques are analyzed and Teredo is found to be the only techniques that works with one or several NAT. The various

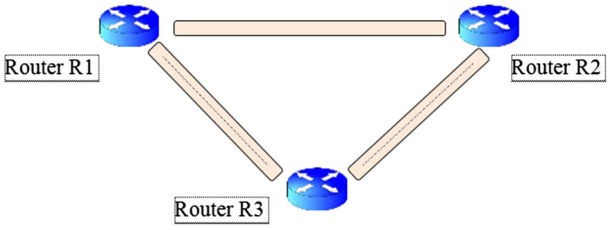


Fig. 1 Static tunnel

tunnels that are studied are Static tunnel, Generic Routing Encapsulation Tunnel, IPv6 in IPv4 Tunnel, 6to4/4to6 tunnel, Tunnel Broker, ISATAP—Intra-Site Automatic Tunnel Addressing Protocol Tunnels, IPv6 Rapid Development Tunnel, Teredo and Multifarious Sym Teredo (MST), which is the proposed tunneling technique [13, 14].

2.1 Static Tunnel

A static configured tunnel is equivalent to a permanent link between two IPv6 domains with the permanent connectivity provided over an IPv4 backbone. This is shown in Fig. 1. If the tunnel partners and the global discovery of the host id disabled, then the only way is to build the static tunnel. Dynamic tunnels will not be formed at that particular moment. The additional reasons for creating static tunnel are that they are given higher priority than dynamics tunnels; they will also allow the user to force a tunnel setup in a situation when something may be preventing automatic tunnel formation.

2.2 Generic Routing Encapsulation Tunnel

GRE tunneling has conventionally been used for encapsulating privately addressed IPv4 datagrams as shown in Fig. 2. In general with convention GRE tunneling, the inner IPv4 addresses are not routable over the network which the GRE tunnel is formed. They are unencrypted tunnels. GRE offers weak authentication. GRE is fundamentally a packaging protocol, envisioned to be able to package any protocols packet into generic data packages that can be carried by other protocol [15].

2.3 IPv6 in IPv4 Tunnel

IPv6 in IPv4 Tunneling uses encapsulation to transmit IPv6 traffic in IPv4 packets and vice versa. This allows for a limited transition where portions of network can drift to IPv6 while the rest of the network remains in its novel state. The IPv6 in IPv4 tunnel is shown in Fig. 3.

2.4 6to4/4to6 Tunnel

6to4/4to6 tunnel is an automatic tunnel. It interconnects isolated IPv6 domains in an IPv4 domain. IPv6 to IPv4 tunnel achieves three functions. Firstly, it allots block of IPv6 space to any host or network that has global IPv4 address [16]. Secondly, it encapsulates IPv6 packets inside IPv4 to communicate, and thirdly it routes the traffic between 6to4 and the

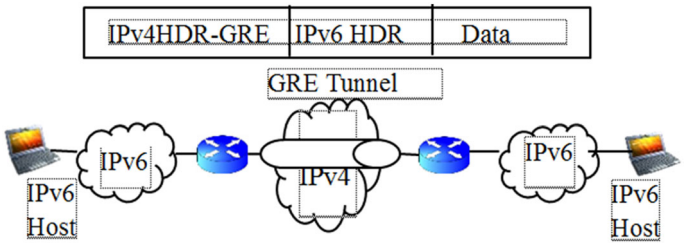


Fig. 2 Generic Routing Encapsulated Tunnel

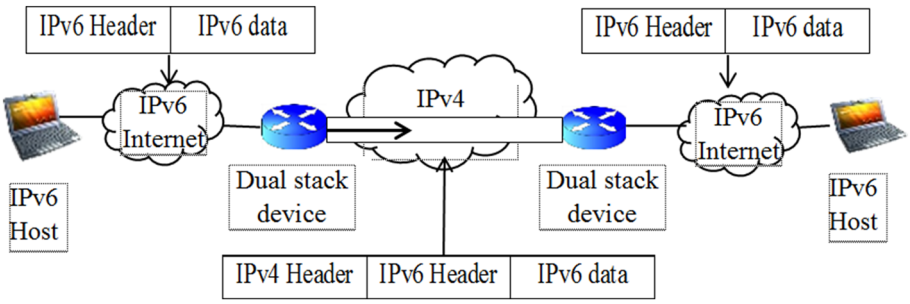


Fig. 3 IPv6 in IPv4 tunnel

native IPv6 network. The 6to4 method is inevitably setup by means of the 2002::/16 IPv6 address space [17]. The IPv4 address for the edge routers is embedded in an IPv6 address that is created. Uses IPv4 embedded address. Embedded IPv4 address allows discovery of tunnel endpoints. 6to4/4to6 tunnel uses relay routers to forward encapsulated IPv6 packets over IPv4 links. The 6to4/4to6 tunnel is shown in Fig. 4.

2.5 Tunnel Broker

IPv6 Tunnel Broker is an alternate method and uses the devoted servers to simplify the establishment of tunnels, these servers are called tunnel brokers. The tunnel broker is accountable for the organization of the tunnels request coming from the end points. This method fits in the scenario with minor, isolated IPv6 sites or IPv6 hosts. These minor sites

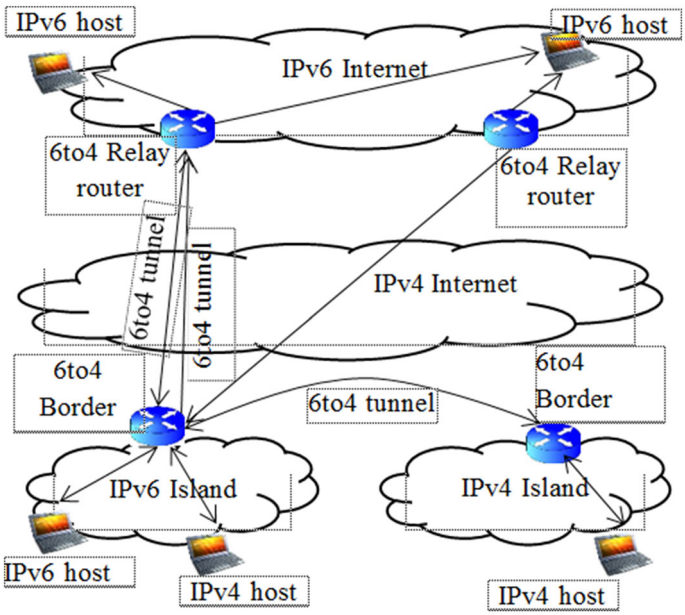


Fig. 4 6to4/4to6 tunnel

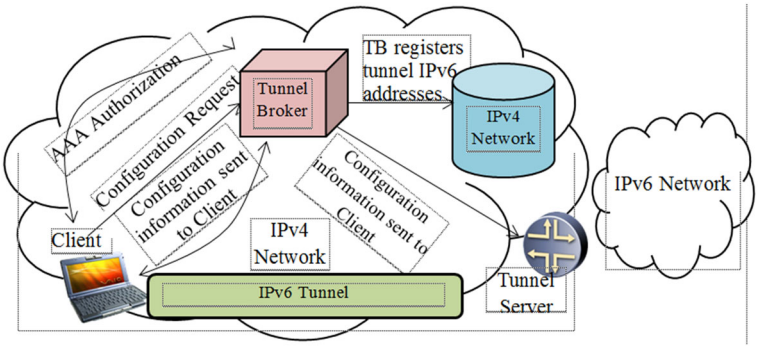


Fig. 5 Tunnel Broker workflow architecture

of isolated hosts are interconnected using today's existing IPv4 based infrastructure [18]. It consists of three basic mechanisms: (1) Clients: Dual-stacked host or router, tunnel end-point, (2) Tunnel Broker: Dual-stacked Internet-connected router, other tunnel end point and (3) Tunnel Server: dedicated server for automatically managing tunnel requests from users, sends requests to Tunnel Server as shown in Fig. 5.

2.6 ISATAP: Intra-Site Automatic Tunnel Addressing Protocol Tunnels

ISATAP allows hosts that are numerous IPv4 hops away from an IPv6 router to participate in the IPv6 network by automatically tunneling IPv6 packet over IPv4 to the IPv6 router as the next hop as shown in Fig. 6. It uses Virtual links to connect IPv6 localities organized within a site that is mainly using IPv4.

2.7 IPv6 Rapid Development Tunnel

The 6rd method was derived from the 6to4 technique but allows the implementer to custom the IPv6 block that was assigned to it. 6rd is a stateless automatically, naturally scalable,

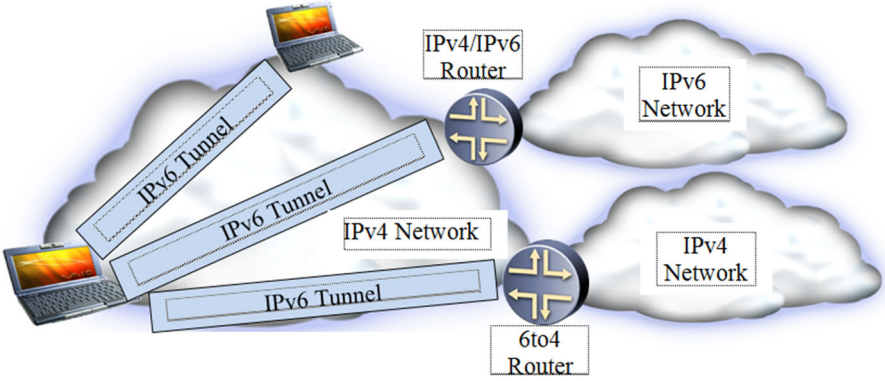


Fig. 6 ISATAP—Intra-Site Automatic Tunnel Addressing Protocol Tunnels

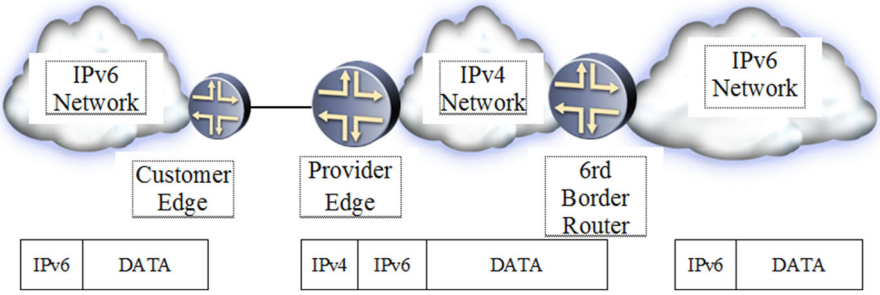


Fig. 7 IPv6 Rapid Development Architecture

resilient, point to multi point tunneling mechanism. Two main hardware components are used: (1) CE—Customer Edge (IPv6 traffic coming from the end user hosts is encapsulated in IPv4 also encapsulated and 6rd traffic received from the Internet through the BR router is de-capsulated). (2) BR—Border Relay (router provides connectivity between the CE routers and the IPv6 network). 6rd approach necessitates customers to have home gateway/routers that can support 6rd and can do the encapsulation of IPv6 packets inside IPv4 and forward them across the Internet backbone. 6rd is a modification of 6to4 [19] as shown in Fig. 7.

2.8 Teredo

Teredo provides address assignment and host to host automatic tunneling for unicast IPv6 traffic when IPv6/IPv4 hosts are located behind one or multiple IPv4 network address translator (NATs). Like 6to4 it uses public relays. Teredo client has access to the IPv4 internet and wants to access to the IPv6 internet. Teredo Server is used to assist the provision of IPv6 connectivity to Teredo clients. A Teredo Relay is an IPv6 router that can receive traffic from the IPv6 internet destined from the Teredo Client and forward it to the Teredo Client Interface. It also accepts packets sent by Teredo clients over their Teredo interface for forwarding to the Ipv6 internet. The Teredo protocol depends on a special IPv6 packet which does not have a payload, called as bubble. This is used to get through the NAT devices. There are two types of bubbles present. Direct bubbles which are sent from Teredo peer to Teredo peer and the indirect bubbles are sent through the Teredo server of the peer.

The Teredo architecture as shown in Fig. 8 consists of Teredo client, Teredo server and a Teredo relay. The Teredo client is an IPv6/IPv4 node which supports a Teredo tunneling interface through which packets are tunneled to either other Teredo client or to the nodes on the IPv6 Internet through the Teredo relay. A Teredo client is made to communicate with the Teredo server to obtain an address prefix from which a Teredo based IPv6 address is configured or to help initiate the communication with the other Teredo clients. A Teredo server is an IPv4/IPv6 node that is connected to both the IPv4 internet as well as to the IPv6 Internet, supporting a Teredo tunneling interface over which packets are received. The Teredo relay is an IPv4/IPv6 router that can forward packets between Teredo clients on the IPv4 internet by using a Teredo tunneling interface and Ipv6-only hosts. Teredo client with an IP address 20.0.0.2 and UDP port number 4096 as shown in the Fig. 8 exchanges the IPv4 UDO messages with the Teredo server to detect the type of NAT and its mapped public IPv4 address as 10.0.0.1 and UDP port number on the NAT as 7863. The mapped addresses along with the port number are encoded in the Teredo client's IPv6 address to

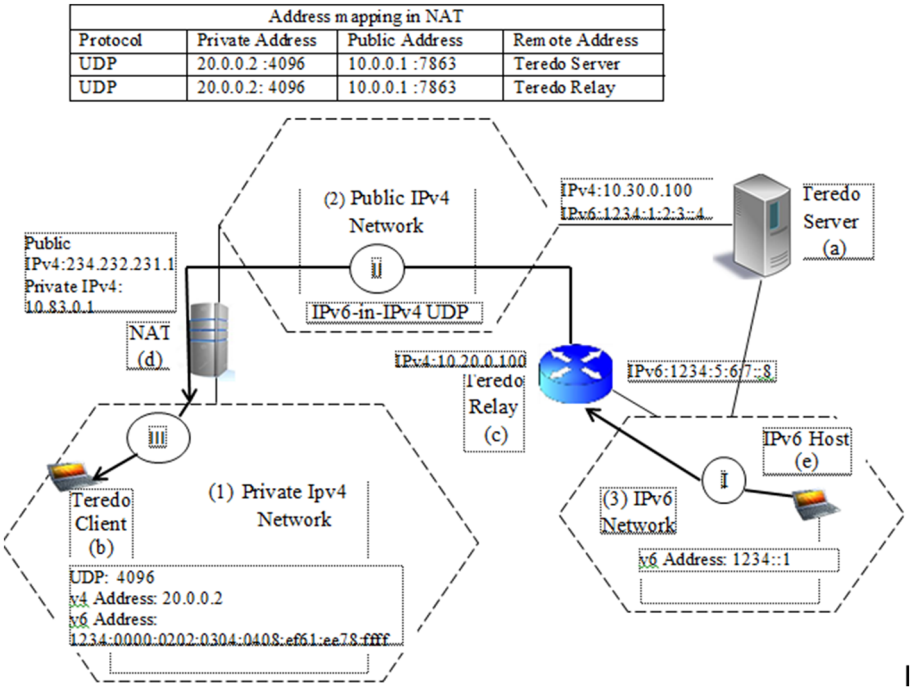


Fig. 8 Architecture of Teredo

identify the NAT. Once the Teredo client has acquired an IPv6 address from the Teredo server, the IPv6 address become known to the hosts in the IPv6 network like dynamic mechanism of Domain name server. A Teredo relay which is nearby to the IPv6 is dynamically preferred based on standard IPv6 routing protocol for IPv6 packets sent from an IPv6 host in an IPv6 network.

The Teredo relay operates an IPv6-in-IPv4 UDP tunnel to sends the IPv6 packets to the Teredo client, in the following way [20]. Firstly the IPv6 host which is located in the IPv6 network tries to send an IPv6 packet to the Teredo client which is present in the private IPv4 network. Secondly, the Teredo relay encapsulates the IPv6 packets in an IPv4 UDP packet. From the destination IPv6 address, the destination IPv4 address along with the port number for the UDP packet is automatically derived, which are the mapped IPv4 address and port number on the NAT for the Teredo client (234.232.231.1 and 7863). The encapsulated packet is sent by the Teredo relay to the NAT. Thirdly, When the NAT present between the public IPv4 network and Private IPv4 network receives the packet; it uses the port number such as 7863 as the key to recover the private IPv4 address 10.0.0.2 and port number 4096 from its mapping table. The NAT then sends the IPv4 UDP packet to the Teredo client which is located in the private IPv4 network.

Upon receiving the packet, the Teredo client located within the private IPv4 network decapsulates the IPv4 UDP packet to obtain the IPv6 packet. The foremost problem with the present Teredo protocol is that, among the four NAT types present [21], Full Cone, Restricted Cone, Port Restricted Cone and Symmetric, the Teredo fails to traverse symmetric NAT alone [22]. The solution proposed in [23], routes IPv6 packets through the

Table 1 Advantages and disadvantages of various tunneling techniques

	Advantages	Disadvantages
Static tunnel	Reliable routing Fault isolation is simpler. Easier troubleshooting Adding more networks does not impact the uptime of existing networks	Complexity in managing configuration of larger networks is high Less resource efficient Less Secure. Since easier to isolate and attack
Generic Routing Encapsulation Tunnel	Tunnel end points can be secured using IPv4 IPsec High-availability is achieved due to mesh topology	Difficult to manage as the number of sites increases the operation effort increases exponentially as due to increase of site as the number of tunnels increases if fully mesh connections are desired More routers, more tunnels cause scaling tedious and time consuming GRE tunnel implementation is rarely available on hosts
IPv6 in IPv4 tunnel	Easier transmission taking into consideration that IPV6 are transmitted as IPV4 Packets Easier to manage and configure tunnels	Overhead traffic is very high Latency is high due to encapsulation and DE capsulation at end points
6to4/4to6 tunnel	Connection of multiple remote IPv6 domains When communicating with IPv6 internet, return path selection is not optimized potential security issue if not secured through IPsec. (Either IPv4 or IPv6)	Number of tunnels supported by the 6to4 router It does not support NAT along the path CPU utilization Hop count is high Fragmentation Issues MTU size Varies RAM increases TTL increases due to processing delay Session time out might occur which results in packet loss in turn it causes retransmission
Teredo	Can be and is used only as a temporary solution Similar to a Dual stack set up Its load balancing design that utilizes a centralized Teredo server for signaling and Teredo relay intended for data packets delivery	Not Compatible with all NAT It can only provide a single IPv6 address per tunnel endpoint. As such, it is not possible to use a single Teredo tunnel to connect multiple hosts, contrary to 6to4 and some point-to-point IPv6 tunnels It cannot be connected to many hosts Relatively less secure Limited Bandwidth
Tunnel Broker	Standalone isolated IPv6 end system It is highly manageability Tunnels setups and managed by ISP	Potential security implication Client tool requires operating through a NAT If broker is topologically remote, round trip time for data may suffer Not well suited if IPv4 address is dynamic Tunnel broker service needs to accept configuration changes remotely, which leads to security implications

Table 1 continued

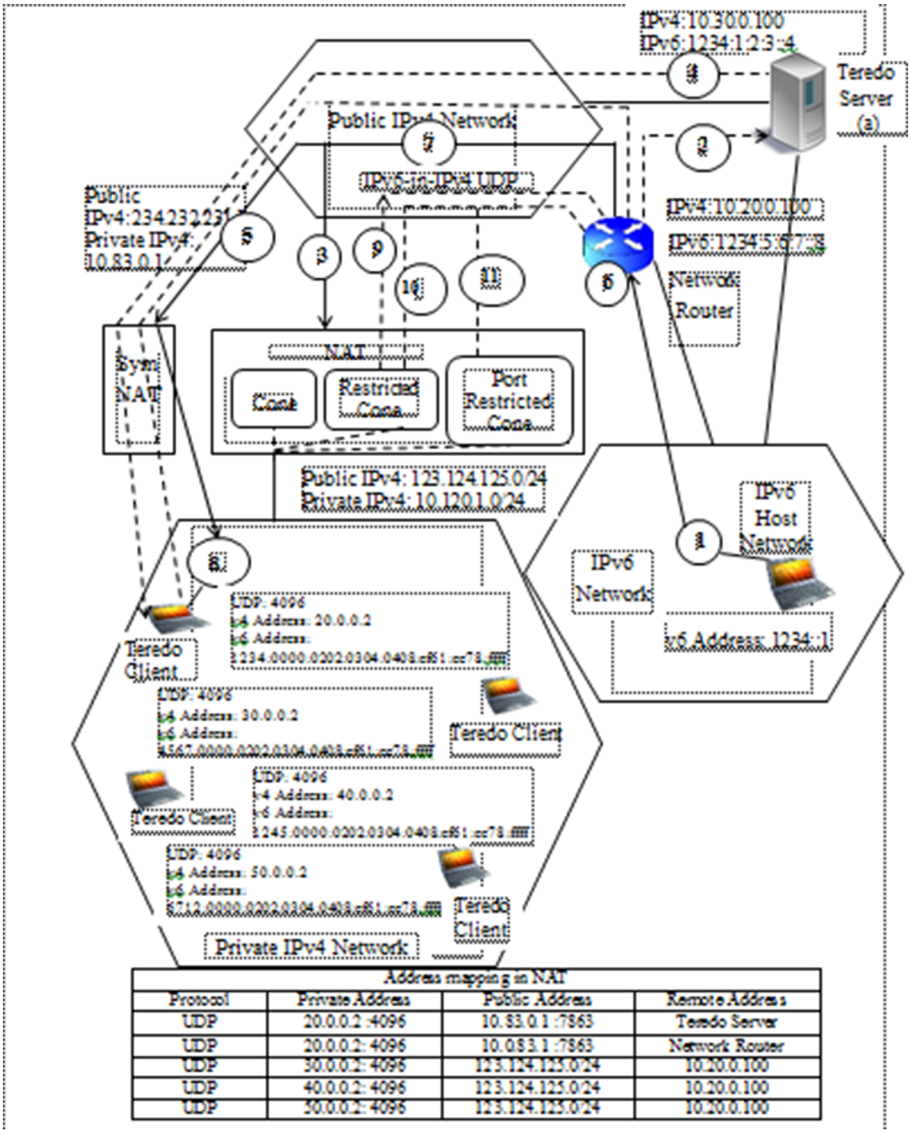
	Advantages	Disadvantages
ISATAP—Intra-Site Automatic Tunnel Addressing Protocol	It is an automatic overlay mechanism It uses underlying IPv4 as broadcast multi-access (NBMA) link Easy to configure and can scale to large number of hosts	Transport layer NAT not supported Delimitation of IPv4 virtual link is required for security reasons No multicast support Can require more setup than other methods Designed for Intra-site use, not Inter-site connectivity
IPv6 Rapid Development Tunnel	Accelerates the deployment of IPv6 to subscribe through the service providers of existing IPv4 network using existing infrastructure and operation It is targeted at unicast IPv6 internet access The subscriber gets native Dual stack IP services IPv6 traffic automatically follows IPv4 6rd border relays are placed at the IPv6 edges and can be addressed through anycast for load balancing and resiliency Transparent for the customer Automatic configuration of the CPE works with public as well as private IPv4 address	It requires upgrading CPE devices Does not provide the complete authentication Overhead is more Address space is wasted as the prefix is allotted even if not explicitly requested Multiple prefix in use on CE's uplink interface Less flexible Local network renumbering if the unique bits of used unique sources changes Change the code running on all the CPEs
Multifarious Sym Teredo tunnel	Ensures that the packet is delivered 100 % Highly flexible Bi-directional communication is ensured Efficient use of routing table Highly scalable	Increase in latency due to various technologies Overhead is high Not well suited if the IP address is dynamic

Teredo server instead of the Teredo relays. Nevertheless, this methodology disrupts the load balancing design of Teredo and it also imposes heavy loading over the Teredo server. After which Sym Teredo was suggested for only one NAT in [24]. Our work proposes MST, which is an extension of both Teredo and Sym Teredo [25] with multifarious symmetric NAT capability. Multifarious Sym Teredo requires trivial amendments to the Teredo relay and Teredo client components. The MST offers backward compatibility with the present Teredo protocol without violating the scattered load balancing strategy [26].

The following Table 1 Show the Advantages and Disadvantages of various tunneling techniques which was observed in the testbed over a Real Time Simulation.

3 Multifarious Sym Teredo Tunneling Technique

The challenges faced during the implementation of the Teredo technique are vulnerability to DoS attack on relay, lack of support of NAT, no transparency in the 6to4 relay which means there is no authenticity of who is using the service, inability to connect to multiple hosts. In order to overcome all the above mentioned challenges, the MST is proposed. The



Address mapping in NAT

Protocol	Private Address	Public Address	Remote Address
UDP	20.0.0.2 :4096	10.83.0.1 :7863	Teredo Server
UDP	20.0.0.2: 4096	10.083.1 :7863	Network Router
UDP	30.0.0.2: 4096	123.124.125.0/24	10.20.0.100
UDP	40.0.0.2: 4096	123.124.125.0/24	10.20.0.100
UDP	50.0.0.2: 4096	123.124.125.0/24	10.20.0.100

Fig. 9 Multifarious Sym Teredo architecture

Fig. 9 describes the architecture of the proposed MST. Similar to the Teredo technique, a Teredo client, Teredo server, Teredo Relay, IPv4/IPv6 hosts (public/private) and router are used to describe the MST architecture. The following briefly describes the MST workflow. The IPv6 address is acquired from Teredo server by the Teredo client, the mapped public IPv4 address and UDP port number are encoded in the IPv6 address. The encoded address will be used by the Network router as the destination of the encapsulated packet. However, a symmetric NAT server will assign diverse mapped port number for every pass through flow. That is, two IPv4 UDP packet sent from the same private IPv4 host to different public

v4 host are interpreted to the same mapped public IPv4 address but dissimilar port numbers. Therefore, the address mapping [27] generated for the flow between the Teredo clients and the Teredo server is dissimilar from that for the flow between Teredo client and Network router.

Unfortunately the mapped public IPv4 address and port number encoded in the Teredo client's IPv6 address will be used by the network router to govern the target address. Hence the network router forwards the IPv6 packets to the Teredo client by means of address mapping deprived of success. To support symmetric NAT traversal [28], Sym Teredo slightly adapts the network router without modifying the Teredo server.

Algorithm for Multifarious Sym Teredo (MST)

Step 1 The host network is the IPv6 network. The IPv6 packet of the host network is directed to a network router (Which is the first hop from the source) is selected rendering to the standard network protocol

Step 2 The network router inspects the packet and checks for the flag status. If flag status equals 1 in the destination clients IPv6 address, then the packet is buffered for future transmission (Step 6) and bubble flag is sent to the Teredo server, the bubble is an IPv6 encapsulated by Ipv4 packet. If the flag status is 0 then go to Step 3

Step 3 The router inspects the destination address in the IPv4 encapsulated IPv6 address and based on the update in the routing table of the router, the packet is forwarded and decapsulated by Cone NAT. Then go to Step 8

Step 4 On receipt of the bubble, Teredo server inserts the source IPv4 address and UDP port number of the network router from which the bubble is originating and then the bubble is mapped to the IPv4 address encoded in the network server client's IPv6 address

The NAT translates this packet and forward it to the client network

Step 5 On receiving the modified bubble, client sends the response to the network router through the symmetric NAT. In this step, the mapping table is created which will be used for the future communication of server and client transmission

Step 6 On receipt of client's response, the network router stores the IPv4 address in the address cache where client's IPv6 address is used as reference key to search the address cache for retrieval. After the network router obtains the address, the previously buffered IPv6 packets (Step 2) is retrieved and kept ready for transmission

Step 7 The IPV6 address is encapsulated in IPv4 address and transmitted as per the updated address cache in the network router. The destination is fetched from the address cache by using the Source address as the key identifier for the transmission

Step 8 The NAT translated the incoming encapsulated packet and send it to the client destination as per the updated address mapping

Step 9 If the client sends a packet to the host network through the Cone NAT interface, it checks the NAT and the packet is forwarded through the network router. Then Go to Step 11

Step 10 If the Client sends a packet through the restricted Cone, Check the routing table for the packet with same destination address. If yes, then go to Step 11. Else discard the packet and go to Step 13

Step 11 Transmit the packet to the network router
Step 12 Decapsulate the packet and deliver it to the source IP address
Step 13 Check if the pack has any port number added to the header of the packet to be sent. Then forward the packet to the Port restricted Cone. If else discard and go to Step 9

4 Testbed Setup Description

Testbed is a platform on which a collection of experimental tools and products that may be organized and are permitted to interact in real-time [29]. Successful tools and products are recognized and are established in an interface in order to have a successful testing. The testbed created for MST proves high scalability and reachability. The MST tunnel connectivity between the IPv4 and IPv6 network is shown in Fig. 10. Cisco 7200 series router which has built in high performance service aggregation, IP to IP gateway support, CDR normalization and quality of service using ToS have been used. The IOS that has been used is c7200-adv enterprise k9-mz.124-15.t8. The other router model used is Cisco 3845 integrated service router. The IOS used is c3845-adv enterprise k9-mz.124-10b. The IOS is loaded to the router by TFTPD32, which is an open source IPv6 ready TFTP server/service. Five cisco routers have been used, in which the R1 acts as a core router. The router R5 is the IPv4 network. The R1 is the IPv6 network. The router R2 along with the switch SW2 and three nodes acts as an asymmetric node. The router R1 along with routers R10 and R11 acts as a Symmetric network. The nodes connected with switch SW4 acts as a source. The Routers R10 and R11 are assigned with both IPv4 and IPv6 address. The link between core router R1 and R10 is a virtual link and the link between core router R1 and R11 is a direct link. The IPv6 address from the source over the switch id encapsulated in IPv4 network comprising of router R5 and R1 are transmitted as per the updated address cache in the

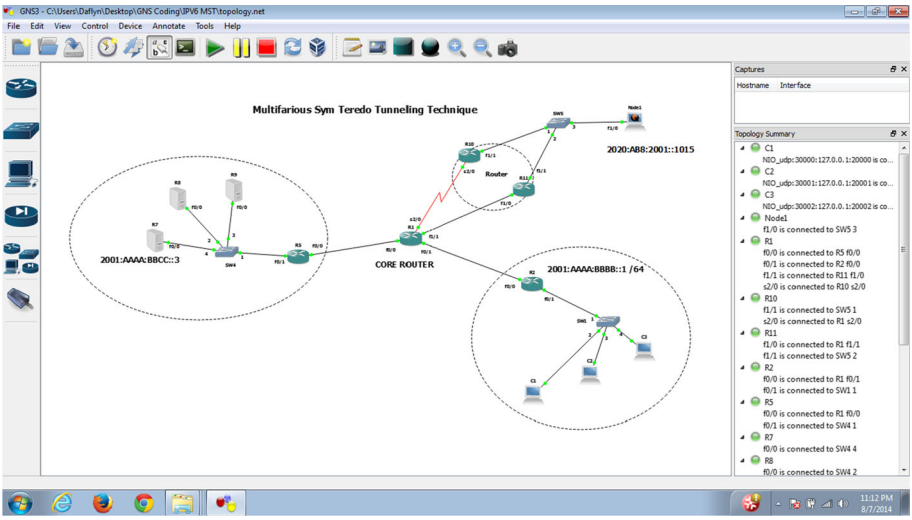


Fig. 10 Multifarious Sym Teredo architecture using Real Time Simulation with Graphical Network Simulator 3

network router R1. The destination is fetched from the source address as the key identifier for the transmission.

The command *ping 2001:aaaa:bbbb::1* is used to connect Source to asymmetric network. In order to trace the route by which the packets are sent, the command trace *2001:aaaa:bbbb::1*. Though the tunnel runs over IPv4 network, it is not visible. The weighted pinging response in IPv6 network is obtained by using the command *ping 2001:aaaa:bbbb::1 − c (number of times) −l (data size)*. The output is obtained by sending packets of different size over the network *n* number of times. The sequence number, time to live and round trip time (RTT) is obtained for various packet size.

5 Real Time Simulation Results

Throughput Analysis is shown in Fig. 11; Throughput which is the number of packets successfully delivered per unit time is controlled by available bandwidth, as well as the available signal-to-noise ratio and hardware limitations (CPU, RAM) [30]. We measured the throughput performance metric in order to find out the rate of received and processed data at the intermediate device (i.e. Router) during the simulation time period. The throughput is calculated from the formula:

$$T_i = \left[P_i / L_i \right] \tag{1}$$

For [i = 1, 2, 3…n].

By Superposition principle the most general equation can be written in the form of additive,

$$T_i = \sum_{i=1}^{n} Pi/Li \tag{2}$$

where, T_i is denoted as the Throughput, P_i is the Packet per Network; L_i is the Latency per Network, i is the Data packets and N is the Total number of the packets in the network. The variations in the total number of packets in the network are proportional to the throughput. The throughput for different packets per network was calculated using the formula below

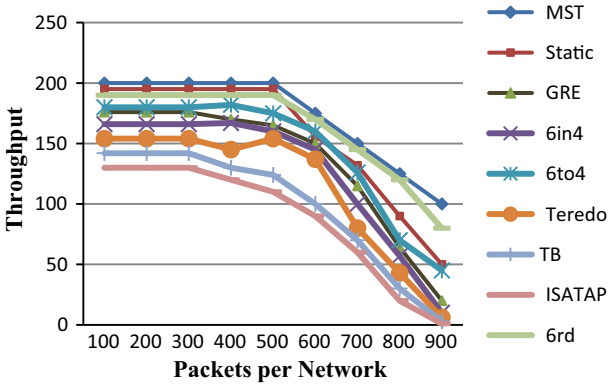


Fig. 11 Test analysis

$$T_i = \left[P_1/L_1 + P_2/L_2 + P_3/L_3 + \ldots + P_N/L_N \right] \quad (3)$$

The threshold Limit taken in the testbed is taken about: 90 % of Link utilization, 75 % of CPU utilization and 75 % of RAM utilization.

We have set up the CPU and RAM utilization threshold as 75 % since there is every chance that the Router as a whole goes down. In order to ensure the continuity of service we have set the limits lower.

The throughput is constant until the CPU utilization is 75 % after which it gradually decreases. Also at the same time throughput is constant until the RAM utilization is 75 % after which it gradually decreases. When the data load keeps on increasing, up to a particular limit based on the capacity of the link, throughput is normal. Beyond the threshold limit the performance (throughput) starts decreasing. Similarly when the number of networks keeps on increasing, up to a specific limit the Router CPU takes care normally. Beyond the threshold limit the performance (throughput) starts decreasing, since the processing load on the CPU increases. Also when the number of networks keeps on increasing, up to a particular limit the Router CPU works steadily normally also as the complexity of the configuration increases the RAM utilization increases. Beyond the threshold limit the performance (throughput) starts decreasing, since the load on the CPU increases.

Round trip time analysis is the response time to identify the quality-of service experienced by the nodes sin IPv6 and IPv4 networks. All nodes on different networks have been involved by means of sending and receiving the ICMP or ICMPv6 packets to each other. The RTT depends on many factors like load at the particular moment of time, Router processor availability and number of virtual routers that are established at that particular point of time. As the complexity of congestion and load increases, the RTT decreases proportionally. With the RTT we can also have a clear idea about the end-to-end cloud loop communication. The RTT is also known as a Ping time and according to, next RTT can be defined by the following calculation which is obtained from the Markov chain theorem.

$$RTT_{next} = (a * RTT_{old}) + ((1 - a) * RTT_{new}) \quad (4)$$

where, a is the smoothing factor (value between 0 and 1).

Figure 12 shows the RTT graph. The RTT is first determined with no load after checking the end to end connectivity. RTT is checked for various tunneling techniques.

Latency Analysis as shown in the Fig. 13 with Samples such as 64 kbps, 128 kbps, 256 kbps, 384 kbps up to 1 Mb+ data are taken and transmitted over the testbed. While testing the data since every packet has to be inspected and a virtual connection has to be established between clients, server and destination, latency increases even before the transmission starts. Whereas in other tunneling techniques the latency obtained is low since there is no virtual connection required before other transmission. Whereas in MTS, all the routing data is stored and decided in the source and destination router, this increases the transmission rate which in turn decreases the latency. There was a major delay in the packet being delivered to the end points. The graph clearly depicts that the performance of static tunneling and MST decreases drastically even on transmission of packets of higher data rates.

Loss Rate Analysis is shown in Fig. 14. In the loss rate analysis, the packet size was increased to measure the corresponding change in the loss rate. Some packets are successfully sent from the client to the server via several network nodes or routers, and some

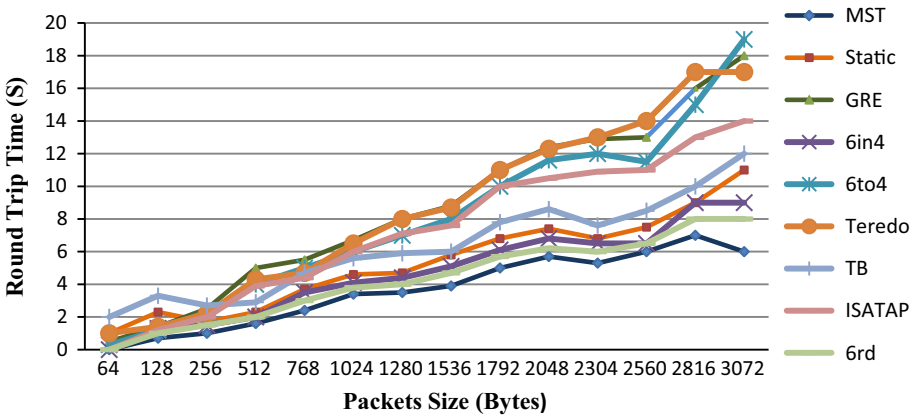


Fig. 12 Round trip time (RTT)

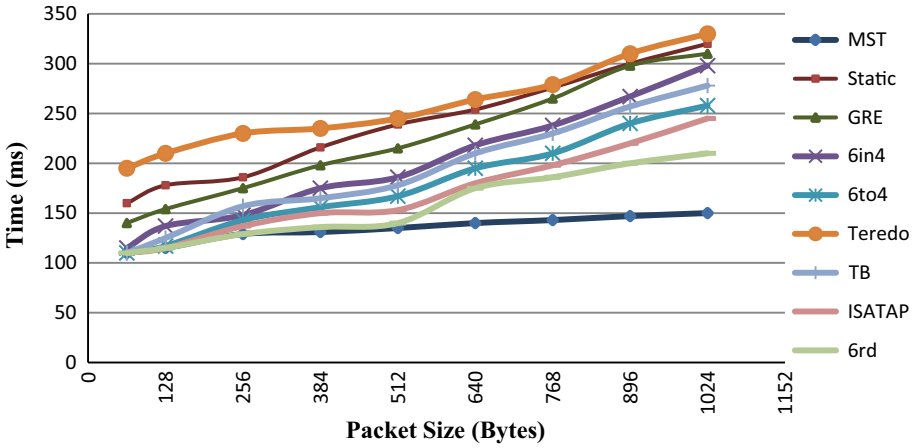


Fig. 13 Latency analysis

packets are lost due to unexpected reasons. In Fig. 14, loss rates analysis for datagram packet size of Nx64 are taken as the samples and transmitted over the testbed. The packet loss is measured in terms of % of packets that are lost. Up to data ratios of 256 kbps there was no significant loss in loss of packets. But when the load increased the packet loss (%) increased considerably. In MST, even though the number of packets sent to various destinations over symmetric and asymmetric NAT networks, the loss rate is very low since the routing architecture is very robust. In the tunneling techniques like Teredo the destination which is located behind a NATed firewall or any other device is lost. In other tunneling techniques like 6to4 and 6in4, though the network traverses through NATed devices due to high overhead some packets are lost.

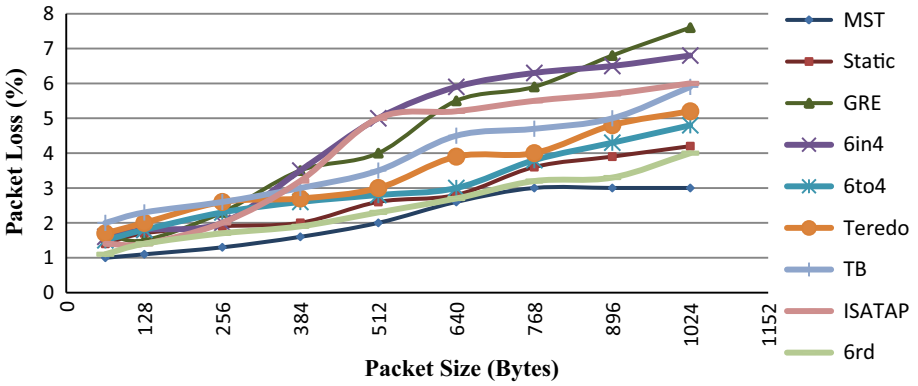


Fig. 14 Loss rate analysis

6 Conclusion and Future Work

This paper describes testbed for MST over a Real time Simulator for IPv4–IPv6 coexistence for tunneling transition techniques. We have achieved a transmission of packets between two different networks with low latency, high throughput, and low data loss over symmetric and asymmetric NATed network by configuring a common Cisco router by making it capable to handle both symmetric and asymmetric NAT. We have achieved low data loss and have ensured that data is delivered at the destination immaterial of the NAT network it travels through. Test analysis was also obtained.

The high availability of the particular network can be ensured by adding one more router at the source and destination in Hot Standby Router Protocol (HSRP) which will help us in increasing the number of nodes in source and destination without any downtime. HSRP is a Cisco proprietary redundancy protocol for establishing a fault-tolerant default gateway. This can be considered as a future work.

References

1. Quynh Anh, N., & Minh Nguyen, N. P. (2012). Transition from IPv4 to IPv6: Best transition method for large enterprise networks.
2. Durand, A. (2001). Deploying ipv6. *Internet Computing, IEEE, 5*(1), 79–81.
3. Phu, N. M., Nguyen, Q. A., Rantapuska, T., Utriainen, J., & Matilainen, M. (2012). Transition from IPv4 to IPv6: The method for large enterprise networks. In *INNOV 2012, the first international conference on communications, computation, networks and technologies* (pp. 5–14).
4. Lee, Y., Durand, A., Woodyatt, J., & Droms, R. (2011). Dual-stack lite broadband deployments following IPv4 exhaustion.
5. Punithavathani, D. S., & Sankaranarayanan, K. (2009). IPv4/IPv6 transition mechanisms. *European Journal of Scientific Research, 34*(1), 110–124.
6. Govil, J., Govil, J., Kaur, N., & Kaur, H. (2008). An examination of IPv4 and IPv6 networks: Constraints and various transition mechanisms. In *Southeastcon, 2008. IEEE* (pp. 178–185). IEEE.
7. Tsirtsis, G. (2000). Network address translation-protocol translation (NAT-PT). *Network*.
8. Aoun, C., & Davies, E. (2007). *Reasons to move the Network Address Translator-Protocol Translator (NAT-PT) to historic status*. RFC 4966, July, 2007.
9. McFarland, S., Sambi, M., Sharma, N., & Hooda, S. (2011). *IPv6 for enterprise networks*. Indianapolis: Pearson Education.

10. Colitti, L., Gunderson, S. H., Kline, E., & Refice, T. (2010). Evaluating IPv6 adoption in the Internet. In A. Krishnamurthy & B. Plattner (Eds.), *Passive and active measurement* (pp. 141–150). Berlin: Springer.
11. Li, C.-S., Lin, F., & Chao, H.-C. (2009). Routing optimization over network mobility with distributed home agents as the cross layer consideration. *Telecommunication Systems, 42*(1–2), 63–76.
12. Azcorra, A., Kryczka, M., & García-Martínez, A. (2010). Integrated routing and addressing for improved IPv4 and IPv6 coexistence. *Communications Letters IEEE, 14*(5), 477–479.
13. Aazam, M., Shah, S. A. H., Khan, I. & Qayyum, A. (2010). Deployment and performance evaluation of Teredo and ISATAP over real test-bed setup. In *Proceedings of the international conference on management of emergent digital ecosystems* (pp. 229–233). ACM.
14. Wang, Y., Ye, S. & Li, X. (2005). Understanding current IPv6 performance: A measurement study. In *Proceedings of the 10th IEEE symposium on computers and communications, 2005. ISCC 2005*. (pp. 71–76). IEEE.
15. Cui, Y., Vautrin, O., Lee, Y., Metz, C., Wu, J., & Wu, P. (2011). Public IPv4 over access IPv6 network.
16. Jayanthi, J. G., & Rabara, S. A. (2010). IPv6 addressing architecture in IPv4 network. In *Second international conference on communication software and networks, 2010. ICCSN'10* (pp. 461–465). IEEE.
17. Cui, Y., Dong, J., Wu, P., Wu, J., Metz, C., Lee, Y. L., et al. (2013). Tunnel-based IPv6 transition. *Internet Computing, IEEE, 17*(2), 62–68.
18. Blanchet, M., & Parent, F., (2010). IPv6 Tunnel Broker with the Tunnel Setup Protocol (TSP).
19. Hamarsheh, A., Goossens, M., & Al-Qerem, A. (2012). Assuring interoperability between heterogeneous (IPv4/IPv6) networks without using protocol translation. *IETE Technical Review, 29*(2), 114–132.
20. Lee, J.-H., Han, Y.-H., Gundavelli, S., & Chung, T.-M.. (2009). A comparative performance analysis on hierarchical mobile IPv6 and proxy mobile IPv6. *Telecommunication Systems, 41*(4), 279–292.
21. Huitema, C. (2006). Teredo: Tunneling IPv6 over UDP through network address translations (NATs).
22. Srisuresh, P., & Egevang, K. (2001) Traditional IP network address translator (Traditional NAT), 1–16.
23. Bao, C., Huitema, C., Bagnulo, M., & Boucadair, M. (2010) *X. Li," IPv6 addressing of IPv4*. IPv6 translators", RFC 6052.
24. Sehgal, A., Talwar, M., Agarwal, A., & Ustuntas, K. (2012) Teredo connectivity between clients behind symmetric NATs. US Patent 8,194,683, issued June 5, 2012.
25. Tsetse, A. K., Wijesinha, A. L., Karne, R. K., & Loukili, A. (2012). A 6to4 gateway with co-located NAT. In *2012 IEEE international conference on Electro/Information Technology (EIT)* (pp. 1–6). IEEE.
26. Zimu, L., Wei, P., & Yujun, L. (2012). An innovative Ipv4–ipv6 transition way for Internet service provider. In *2012 IEEE symposium on robotics and applications (ISRA)* (pp. 672–675). IEEE.
27. Lim, T. M., Lee, B.-S., Yeo, C. K., Tantra, J. W., & Xia, Y. (2009). A terminal-assisted route optimized NEMO management. *Telecommunication Systems, 42*(3–4), 263–272.
28. Mrugalski, T., Wozniak, J., & Nowicki, K. (2013). Dynamic host configuration protocol for IPv6 improvements for mobile nodes. *Telecommunication Systems, 52*(2), 1021–1031.
29. Sailan, M. K., Hassan, R., & Patel, A. (2009). A comparative review of IPv4 and IPv6 for research test bed. In *2009 International Conference on electrical engineering and informatics*, 5–7 August 2009, Selangor, Malaysia.
30. Lopes, N. V., Nicolau, M. J., & Santos, A. (2013). A QoS-enabled resource management scheme for F-HMIPv6 micro mobility approach. *Telecommunication Systems, 52*(1), 341–357.

Sheryl Radley completed Bachelor of Engineering in Electronics and Communication Engineering, in 2007 from Karunya Institute of Technology and sciences. She has completed Master of Engineering in Communication Systems, in 2009 from National Engineering College, Kovilpatti. She is a Full time Research Scholar under Anna University Chennai; she is currently doing the research in IPv4–IPv6 Transition Techniques in Government College of Engineering, Tirunelveli, Tamil Nadu. Her main interest and work areas are mobile computing and Networking.

D. Shalini Punithavathani completed her Bachelor of Science, in 1979 in Sarah Tucker College, Tirunelveli, India affiliated to Madurai Kamarajar University, Tirunelveli, India. She completed her Bachelor of Technology in Electronics, in 1982 from Madras Institute of Technology, Chennai, India affiliated to Anna University, Chennai, India. She completed her Master of Engineering in Computer Science and Engineering, in 1990 from Government College of Technology, Coimbatore, India affiliated to Bharathiar University, Coimbatore, India. She completed her Doctorate in Philosophy, entitled "Study and Implementation of IPv4 to v6 translation techniques" in 2010 in Anna University, Chennai, India. She is working as a principal in Government College of Engineering, Tirunelveli, Tamil Nadu. Her main interest and work areas are mobile computing and Networking.