IPv4 TO IPv6 Transition Using Windows OS

Shubhangi Kharche[1] and B D Biranale[2]

***Abstract* – Internet Protocol version-4 (IPv4) to Internet Protocol version-6 (IPv6) transition is necessary in view of fast expanding internet usage and increased demand on the IP address space. Network Address Translation (NAT) has temporarily solved the issue of limited address availability. The transition process is complex, as it has to deal with issues related to IPv4 - IPv6 interoperability including routing, DNS error handling, router configuration etc. In this paper, a solution named ISATAP (Intra-site Automatic Tunnel Addressing Protocol) on Windows operating system and dual stack approach is presented. ISATAP is used to deploy unicast IPv6 connectivity on an existing IPv4 networking environment. ISATAP using Windows platform uses a tunneling approach to transport IPv6 traffic across an existing IPv4 infrastructure. IPv6 packets are encapsulated with an IPv4 header to achieve an efficient solution for transition**

***Index Terms* – Encapsulation , ISATAP, IPv6 tunneling, serverless auto-configuration, transition.**

I. INTRODUCTION

Over the last decade, Internet Engineering (IETF) has been working on the deployment of IPv6 [1, 2] to replace the current IPv4 [3] protocol. Recent attention to IPv6 has increased immensely because of IPv4 address shortages, mobility requirements, and the need for global, secure, seamless, and permanent connectivity. The next generation Internet that uses IPv6 promises to enable a whole new breed of applications. A common question about IPv6 is: "Does one have to upgrade their whole environment end-to-end to support IPv6 before an application can utilize IPv6 connectivity"? The answer is "no." The IETF IPng Transition Working Group has been working on several transition strategies, tools, and mechanisms. An IPv6 transition technology named ISATAP allows you to deploy unicast IPv6 connectivity on an existing IPv4 networking environment without making many changes to the infrastructure. ISATAP uses a tunneling approach to transport IPv6 traffic across an existing IPv4 infrastructure. IPv6 packets are encapsulated with an IPv4 header. This approach allows organizations to embark on the IPv6 integration journey without having to spend a large amount of time and financial resources to upgrade their infrastructure immediately to support native IPv6 services. In general, all transition mechanisms provided by IETF encapsulate IPv6 packets into IPv4 packets, and transport them over an IPv4 network infrastructure. We expect to rely on these transition strategies as the Internet shifts from the traditional IPv4 to an IPv6-based internet while retaining both IPv4 and IPv6 throughout the transition phase.

The main goal of this work is to demonstrate transition mechanism, namely ISATAP. We have implemented a pilot IPv6 network using five computers, two of which are acting like clients, two others as routers, and one as a DNS server. We have shown serverless auto-configuration in the client computers as well as routing of IPv6 packets over an IPv4 network and ISATAP routing.

II. IPV4 TO IPV6 TRANSITION MECHANISMS

Some currently available transition mechanisms are: Dual Stacks [4], DTI & Bump in-dual-stack, NAT Protocol Translator [5], Stateless IP / ICMP Translator (SIIT), Assignment of IPv4 Global Addresses to IPv6 Hosts (AIIH), Tunnel Broker [6], 6 to 4 mechanism [7], *6*-over-4 mechanism [8], and IPv6 in IPv4 tunneling [9].

Dual Stacks are easiest to implement, however complexity increases at the hosts due to both infrastructures and the cost is higher due to a quite complex technology stack. NAT Protocol Translator has scaling and DNS issues, and has single point of failure disadvantage. The Tunnel Broker dynamically gains access to tunnel servers, but has authentication and scaling issues. 6-to-4 mechanism creates dynamic stateless tunnels over IPv4 infrastructure to connect 6-to-4 domains. 6-over-4 mechanism allows the interconnection of isolated IPv6 hosts to connect over the IPv4 infrastructure without requiring IPv6 enabled routers or explicit tunnels. IPv6 in IPv4 tunneling allows existing infrastructure to be utilized via manually configured tunnels.

We have chosen to pursue the *IPv6 in IPv4 tunneling* and *6-over-4* as a transition mechanism. 6-over-4 mechanism performs the encapsulation at the host, and therefore known as *host-to-host encapsulation*. The IPv6 in IPv4 tunneling performs the encapsulation at the routers, and hence is known as *router-to-router tunneling.* The router-to-router tunneling enables two entire LANs to be upgraded to IPv6, while maintaining connectivity to the rest of the Internet. Host-to-host encapsulation is also addressed mainly because of its simplicity of implementation, and offers another method of making the transition from IPv4 to IPv6 as smooth as possible.

Encapsulation of IPv6 packets within IPv4 packets allows two IPv6 hosts/networks to be connected with each other while running on existing IPv4 networks. IPv6 packets are

[1] Shubhangi Kharche is a Lecturer, Department of Electronics and Telecommunication SIES Graduate School of Technology, Nerul, Navi Mumbai. Email: shubhangi.kharche@gmail.com

[2] Dr B D Biranale is Head, Department of Computer Engineering, Principal, Brahmdevdada Mane Institute of Technology, Solapur. Email: principal@bmitsolapur.com, bdbiranale@yahoo.co.in

encapsulated in IPv4 packets and then are transmitted over IPv4 networks like ordinary IPv4 packets. At the destination, these packets are de-capsulated to the original IPv6 packets. It should be noted that while encapsulation of IPv6 packets in IPv4 packets, only IPv4 routing properties will be utilized and hence the IPv6 packet will lose any special IPv6 features until it is de-capsulated at the receiving host/router.

III. IPv6 TUNNELING

There are two types of tunnels [10] in IPv6: a) Automatic tunnels and b) Configured tunnels. Automatic tunnels are configured by using IPv4 address information embedded in an IPv6 address – the IPv6 address of the destination host includes information about which IPv4 address the packet should be tunneled to. Configured tunnels must be configured manually. These tunnels are used when using IPv6 addresses that do not have any embedded IPv4 information. The IPv6 and IPv4 addresses of the endpoints of the tunnel must be specified.

Tunneling provides a way to use an existing IPv4 routing infrastructure to carry IPv6 traffic. The key to a successful IPv6 transition is compatibility with the existing installed base of IPv4 hosts and routers. Maintaining compatibility with IPv4 while deploying IPv6 streamlines the task of transitioning the Internet to IPv6. While the IPv6 infrastructure is being deployed, the existing IPv4 routing infrastructure can remain functional, and can be used to carry IPv6 traffic.

IPv6 or IPv4 hosts and routers can tunnel IPv6 datagrams over regions of IPv4 routing topology by encapsulating them within IPv4 packets. Tunneling can be used in a variety of ways as shown in Table 1 below:

TABLE 1: TUNNELING TECHNIQUES

Router-to-Router	IPv6 or IPv4 routers interconnected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. In this case, the tunnel spans one segment of the end-to-end path that the IPv6 packet takes.
Host-to-Router	IPv6 or IPv4 hosts can tunnel IPv6 packets to an intermediary IPv6 or IPv4 router that is reachable through an IPv4 infrastructure. This type of tunnel spans the first segment of the packet's end-to-end path.
Host-to-Host	IPv6 or IPv4 hosts that are interconnected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. In this case, the tunnel spans the entire end-to-end path that the packet takes.
Router-to-Host	IPv6/IPv4 routers can tunnel IPv6 packets to their final destination IPv6 or IPv4 host. This tunnel spans only the last segment of the end-to-end path.

Tunneling techniques are usually classified according to the mechanism by which the encapsulating node determines the address of the node at the end of the tunnel. In router-to-router or host-to-router methods, the IPv6 packet is being tunneled to a router. In host-to-host or router-to-host methods, the IPv6 packet is tunneled all the way to its final destination.

The entry node of the tunnel (the encapsulating node) creates an encapsulating IPv4 header and transmits the encapsulated packet. The exit node of the tunnel (the decapsulating node) receives the encapsulated packet, removes the IPv4 header, updates the IPv6 header, and processes the received IPv6 packet. However, the encapsulating node needs to maintain soft state information for each tunnel, such as the maximum transmission unit (MTU) of the tunnel, to process IPv6 packets forwarded into the tunnel.

IV. TRANSITION USING ISATAP

ISATAP utilizes an innovative principle in which an IPv4 network emulates a logical IPv6 subnet to a set of ISATAP hosts. This principle allows all ISATAP nodes, no matter where they are located on the IPv4 network, to automatically tunnel to each other for IPv6 connectivity and reach other IPv6-capable networks or the IPv6 Internet through an ISATAP router. The benefits of ISATAP are the following:

An existing IPv4 infrastructure can provide [11] unicast IPv6 connectivity immediately with the only requirement being the configuration of an ISATAP router.

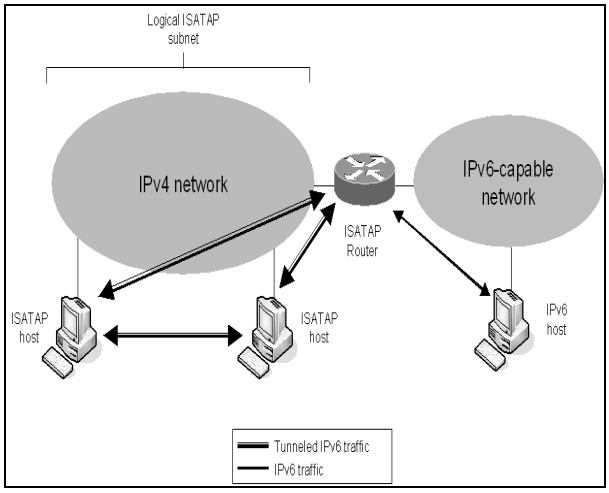

Figure 1: ISATAP Tunneling between ISATAP hosts and through an ISATAP Router.

Native IPv6 capabilities can be enabled first in the backbone, while allowing other parts of the IPv4 infrastructure to preserve their investment and naturally evolve to support native IPv6. ISATAP islands can be created to allow gradual evolution to native IPv6 capabilities within different parts of an organization without blocking end-to-end IPv6 service deployments. End-to-end IPv6 services can be enabled and maintained using ISATAP while allowing access to native IPv6 infrastructure, such as a native IPv6 backbone or the IPv6 Internet.. Table2 Compares the components to be upgraded to support an ISATAP-based IPv6 deployment.

TABLE2: COMPONENTS TO BE UPGRADED TO SUPPORT AN ISATAP-BASED IPV6 DEPLOYMENT.

Infrastructure Component	Currently Deployed In an IPv4 Infrastructure	Upgrade Needed to Support IPv6 Using ISATAP
DNS	Yes	Possibly*
Active Directory	Yes	None
DHCP	Yes	None
IPsec	Yes	None**
Routers	Yes	None
Switches	Yes	None
Firewalls/NAT	Yes	None***
ISATAP router	No	ISATAP router
VPNs	Yes	None

* The DNS servers must be able to support registration and querying for IPv6 AAAA resource records. DNS servers are also required to support DNS queries over IPv6.

** The ISATAP router must either support IPsec or it must be exempted from IPsec traffic protection.

*** ISATAP traffic cannot traverse firewalls or NATs that are transport layer port dependent (such as TCP, UDP, etc.).

The IPv6 transition is a gradual process, and it needs a staged transition plan, especially for large organizations. The solution offered by ISATAP enables IPv4 applications to run in IPv6 networks without any modification and/or recompilation. This allows one to take advantage of abundant IPv4 applications when deploying IPv6, thus helps speeding up its deployment. In addition, IPv4 legacy applications can still run after some network infrastructure is upgraded to IPv6, thus user's present investment can be protected without hampering the IPv6 deployment. In general, deploying ISATAP within the present IPv4 networks includes the following three stages:

1) IPv4 routing: In this stage IPv4 connectivity and the automatic configuration of local link address is obtained
2) IPv6 routing: In this stage all nodes can be reached using IPv6 traffic.
3) ISATAP routing. ISATAP is an address assignment and automatic tunneling technology that is used to provide unicast IPv6 connectivity between IPv6/IPv4 hosts over an IPv4 intranet. This phase removes IPv6 connectivity for Subnet 2 and Subnet 3 and restores it using ISATAP as shown in figure 2.

V. IMPLEMENTATION OF THE PROTOTYPE SYSTEM ON WINDOWS OS

For end-to-end IP communication between IPv4 and IPv6 hosts, the ISATAP technique is used to encapsulate IPv4 packets in IPv6 to travel in the IPv6 network based on IPv6 routing. Border routers serve as tunnel endpoints and forward IPv6 packets to IPv4 networks and vice versa. When a border router receives tunnel packets, it decapsulates the tunnel header and forward them to IPv4 networks. On the other hand, when a border router receives IPv4 packets, it encapsulates them with the IPv6 tunneling header and transmits to the IPv6 host. The encapsulation effectively suppresses the direct movement of IPv4 packets within an IPv6 network, and instead allows their movement based only on IPv6 routing tables, thereby simplifying the IPv6 network management.

Within the IPv6 network, each dual stack host communicates only in IPv6 or IPv6-encapsulated packets. An IPv4 application running on an IPv6 node uses either a private IPv4 address to talk to another IPv4 application within the IPv6 network or use IPv4 address to talk to another IPv4 application in the Internet. Since only IPv6 packets are allowed within the IPv6 network, all private-address or public-address IPv4 packets within the IPv6 network must be encapsulated with IPv6 headers with the help of dual stack approach ISATAP technique. Figure 2 shows ISATAP with its components

- Dual stack host (Client1, Client2): a host in an IPv6 network with both IPv4 and IPv6 stacks. Both IPv4 and IPv6 based applications can run on this host.
- DNS server (DNS): BIND server providing normal DNS functions that can resolve both private IPv4 addresses and IPv6 addresses for the IPv6 network.
- Border router (Router1, Router2): a dual stack router sitting on the boundary between an IPv6 and an IPv4 network. The router maintains an IPv6-IPv4 address with static routes for each outbound and inbound packet.

The following typical communication scenario is considered:

IPv6-to-IPv6 scenario; the communication from a dual stack node to another dual stack node in two different IPv6 networks over an IPv4 network, e.g., Client2 to Client1 via router Router1& Router2 as in figure 2.

ISATAP described in this paper has been implemented on Windows 2003 server with service pack 1 (SP1) as Routers, DNS server and Windows XP with service pack 2 (SP2) as clients

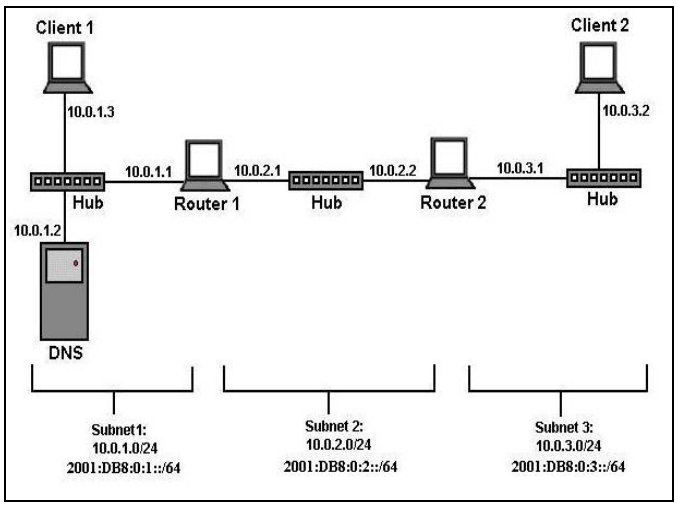

Figure 2: Network diagram of IPv6 testbed

We have setup an IPv6testbed as described in figure 2 within the departmental intranet and deployed ISATAP prototype system in the IPv6 network. On our dual stack Windows nodes in the IPv6 network, we used available IPv4 applications to communicate with the Internet. We could visit dual stack hosts within our IPv6 test-bed from the Internet seamlessly. None of IPv4 programs required any modification and/or recompilation to enable them running in the IPv6 network. We have tested some basic network applications such as telnet, ftp, email, WWW, and we also ran IPv4 version X-windows program from our IPv6 test-bed to access the Windows servers on the IPv4 network. The implementation has successfully demonstrated that the ISATAP can provide a transparent communication between IPv6 and IPv6 nodes over an IPv4 network, as well as guarantee end-to-end IP connection.

In general deploying ISATAP within the present IPv4 network include the following steps:

On all Router1, Router2, DNS server, Client1 and Client2 IPv6 are installed.

Step1: Upgrade all nodes within a chosen subnet to dual stacks. Configure Client1 as a client computer.
Install Windows XP Professional with SP2 as a workgroup computer. Set the Administrator password. After restarting, log on as Administrator.

Install Ipv6 by typing: *netsh interface ipv6 install* at command prompt

Configure the TCP/IP protocol with the IP address of 10.0.1.3, the subnet mask of 255.255.255.0, a default gateway of 10.0.1.1, and the DNS server IP address of 10.0.1.2.
Similarly configure Client2 with IP addresses as shown in figure 2

Step2: Configure servers Router1 and Router2 as routers between Subnet1, Subnet2 and Subnet2, Subnet3 respectively
Install Windows Server 2003 with SP1, Standard Edition, as a workgroup computer.
Set the Administrator password.
After restarting, log on as Administrator.
At the command prompt, install the IPv6 protocol by typing: *netsh interface ipv6 install*

In Control Panel-Network Connections, rename the LAN connection connected to Subnet 1 to *Subnet 1 Connection* and rename the LAN connection connected to Subnet 2 to *Subnet 2 Connection.*

For *Subnet 1 Connection*, configure the TCP/IP protocol with the IP address of 10.0.1.1, the subnet mask of 255.255.255.0, and the DNS server IP address of 10.0.1.2.

For Subnet 2 Connection, configure the TCP/IP protocol with the IP address of 10.0.2.1, the subnet mask of 255.255.255.0, and a default gateway of 10.0.2.2.

Start the registry editor (Regedit.exe) and set HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\ Services\Tcpip\Parameters\\IPEnableRouter to 1.
This step enables IPv4 routing between Subnet1 and Subnet2.
Restart the computer.
Similarly configure Router2 with the IP addresses shown in figure 2 to enable routing between Subnet2 and Subnet3

Step3: Create a static IPv6 routing infrastructure so that Client1 can be reachable from Client2 using IPv6 traffic

On Router1, type the *ipconfig* command to obtain the link-local addresses of the interfaces connected to Subnet 1 Connection and Subnet 2 Connection.

On Router2, type the *ipconfig* command to obtain the link-local addresses of the interfaces connected to Subnet 2 Connection and Subnet 3 Connection.
On Router1, type the following commands:
netsh interface ipv6 set interface "Subnet 1 Connection" forwarding=enabled advertise=enabled
netsh interface ipv6 set interface "Subnet 2 Connection" forwarding=enabled advertise=enabled
netsh interface ipv6 add route 2001:db8:0:1::/64 "Subnet 1 Connection" publish=yes
netsh interface ipv6 add route 2001:db8:0:2::/64 "Subnet 2 Connection" publish=yes
netsh interface ipv6 add route ::/0 "Subnet 2 Connection" nexthop=Router2AddressOnSubnet2 publish=yes

In the preceding command, *Router2AddressOnSubnet2* represents the link- local address assigned to the Subnet2 Connection interface on Router2.

On Router2, type the following commands:
netsh interface ipv6 set interface "Subnet 2 Connection" forwarding=enabled advertise=enabled
netsh interface ipv6 set interface "Subnet 3 Connection" forwarding=enabled advertise=enabled
interface ipv6 add route 2001:db8:0:2::/64 "Subnet 2 Connection" *publish=yes*
netsh interface ipv6 add route 2001:db8:0:3::/64 "Subnet 3 Connection" publish=yes

netsh interface ipv6 add route ::/0 "Subnet 2 Connection" nexthop=Router1AddressOnSubnet2 publish=yes

Step4: Verify the IPv6 routing structure.
On Client1, type the *ipconfig* command to check for a new global IPv6 address as in figure 3

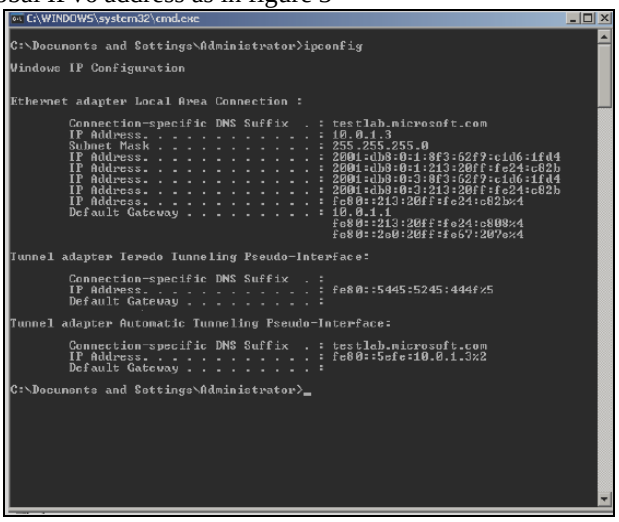

Figure 3: IPv6 Routing Structure

On Client2, type the following commands:

ping Client1GlobalAddress

tracert -d Client1GlobalAddress

A successful ping and tracert as in figure 4 demonstrates that IPv6 static routes have been created and are functioning.

You can view the entries in the Router1 neighbor cache for Client1 and Router2, by typing the following on Router1:

netsh interface ipv6 show neighbors

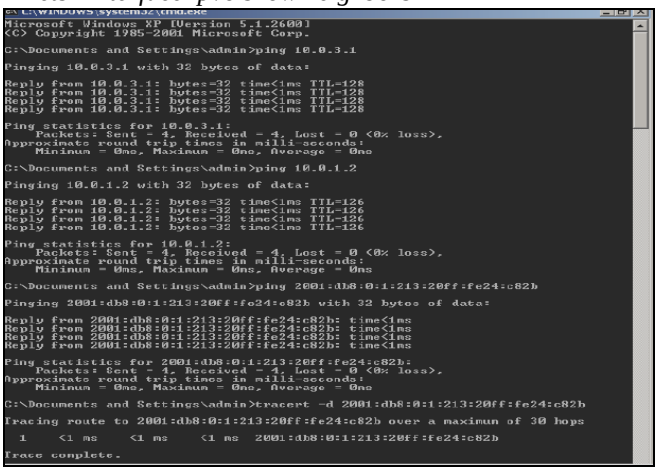

Figure 4: Ping and Tracert Results

Step 5: Configure DNS to resolve name to IPv6 address to resolve host names to IPv6 addresses, you must first configure DNS. On DNS1, create an AAAA record for Client2 as seen in figure 5 with the DNS name client2.testlab.microsoft.com for its global IPv6 address using the IPv6 host resource record type.

On Client2, type the following command:

ping client1

A successful ping demonstrates that host names can resolve to IPv6 addresses.

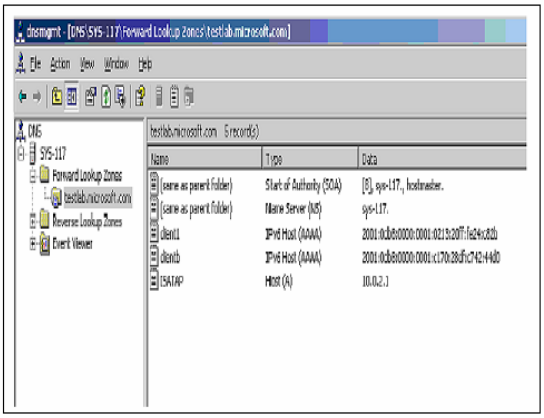

Figure 5: DNS A & AAAA records

VI. CONCLUSION

Thus we have successfully demonstrated an easy and efficient IPv4 to IPv6 transition solution using ISATAP for achieving dual stack IPv6 nodes to communicate over an IPv4 network. Compared with other related methods of transition, ISATAP has several obvious advantages:

Transparent end-to-end IP communication, scalable deployment, and most importantly, running of IPv4 applications in an IPv6 environment seamlessly. We have implemented a prototype system and tested its functions in our IPv6 testbed using Windows Platform. We believe ISATAP as an IPv4 to IPv6 transition solution has great potential to speed up IPv6 deployment without much change in infrastructure.

VII. REFERENCES

[1] Deering, S, and Hinden, R, "Internet protocol, version 6 (IPv6) specifications", RFC2460, December 1998

[2] Gilligan, R and Nordmark, E, "Transition mechanisms for IPv6 hosts and routers", RFC2893, August 2000

[3] Kitamura, H, Jinzaki, A, and Kobayashi, S, "A SOCKS-based IPv6/IPv4 gateway mechanism", Internet Draft (draft-ietf-ngtranssocks-gateway-06.txt)

[4] Hinden, R and Deering, S, "IP version 6 addressing architecture", RFC2373, July 1998

[5] Crawford, M and Huitema, C, "DNS extensions to support IPv6 address aggregation and renumbering", RFC2874, July 2000

[6] Conta, A and Deering, S, "Internet control message protocol (ICMPv6) for the Internet Protocol ver 6 (IPv6) specifications", RFC2463, December, 1998

[7] Conta, A and Deering, S, "Generic packet tunneling in IPv6 specification", RFC2473, December 1998

[8] Bound, J, Toutain, L, Afifi, H, Dupont, F and Durand, A, "Dual stack transition mechanism (DSTM)", Internet Draft (draft-ietf-ngtrans-dstm-04.txt)

[9] Rekhter, Y, Moskowitz, B, Karrenberg, D, de Groot, GJ and Lear, E, "Address allocation for private internets", RFC1918, February 1996

[10] HP-UX IPv6 Transport Administrator's Guide: HP-UX 11i v2 http://docs.hp.com/en/B2355-90795/ch06s02.html?btnPrev=%AB%A0prev, November 2007

[11] Manageable Transition to IPv6 using ISATAP http://www.microsoft.com/ipv6

VIII. BIOGRAPHIES.

Shubhangi Kharche was born in Chandigarh on August 24, 1977. She graduated from Government Engineering College, Aurangabad, completed Post Graduation from Jawaharlal Nehru Engineering College, Aurangabad, affiliated to Dr. Babasaheb Ambedkar Marathwada University, Aurangabad. She is a **CCNA** (Cisco Certified Network Associate). Her employment experience includes Babasaheb Naik Engineering College, Pusad, Marathwada Institute of Technology, Bulandshahr, Godavari College of Engineering, Jalgaon, SIES Graduate School of Technology, Nerul. She is a member of ISTE, IEEE and IETE.

Dr B D Biranale was born in Ganeshwadi near Miraj on June 2, 1953. He graduated from Walchand College of Engineering, Sangli in 1977, completed M Tech (Prodn Engg) from IIT, Bombay in 1981 and was awarded Ph D (Comp Engg) by NIT Karnataka, Surathkal in March 2006. He has served Gogte Institute of Technology, Belgaum, as Assistant Professor, Padmashri Dr Viththalrao Vikhe Patil College of Engineering, Ahmednagar, Marathwada Institute of Technology, Aurangabad and Bulandshahr, as Professor and Heads of Department. He is now working as the Principal at Brahmdevdada Mane Institute of Technology, Solapur. So far, he has presented / published 12 technical papers, out of which four are at International level. He is a life member of ISTE, New Delhi.