Proceedings of The National Conference
On Undergraduate Research (NCUR) 2008
Salisbury University
Salisbury, Maryland
April 10 - 12, 2008

Lessons Learned in Transitioning from Internet Protocol Version 4 to Internet Protocol Version 6

Joshua E Domagalski
Computer Science
United States Naval Academy
Annapolis, MD

Faculty Advisor: Dr. Patrick Vincent

Abstract

Internet Protocol Version 6 (IPv6) has been proposed as a replacement to the current networking protocol, IPv4, used in the global Internet. IPv6 represents an entirely new protocol, incorporating improved routing capability, enhanced ability to support real-time audio and video traffic, better security and privacy, as well as a much larger address space. Technical challenges and operational requirements have hindered the Department of Defense from investigating and testing IPv6 on a large scale. To help facilitate the conversion to IPv6, we have built, tested, operated and maintained a pilot IPv6 network between the United States Naval Academy in Annapolis, Maryland and the United States Military Academy in West Point, New York. This network hosted multiple operating systems and employed a Domain Name System server, a web server, a File Transfer Protocol server, a Dynamic Host Configuration Protocol server, in addition to multiple PC and Unix-based clients. This network was then used to provide validation and refutation of operational concepts developed for the transition from IPv4 to IPv6. This research included investigations of network management, address allocation, Domain Name Services, protocol capabilities and communication, as well as transition techniques for migrating from the IPv4 protocol to IPv6. We present the results of our experience in deploying and testing an IPv6 network, discuss proposed best practices for the utilization of IPv6, and provide a list of lessons learned to include various incompatibility issues that were noted with many popular software applications. Finally, in exploring the added quality of service capability provided in IPv6, we present our preliminary results in testing and analyzing Voice over Internet Protocol on our IPv6 network.
Keywords: IPv6, IPv4, DoD

1. Introduction

Originally, the Department of Defense was a driving force in the creation and implementation of information technologies. However, as the economy and market have developed at an exponential rate, security considerations, technical challenges, and operational requirements have hindered the Department of Defense (DoD) from investigating, testing, and implementing many of these new information technologies. Thus, the DoD is now finding itself in a position where it is forced to change to maintain compatibility rather than driving the change itself.

Internet Protocol Version 6 (IPv6) has been proposed as a replacement to the current networking protocol, IPv4, used in the global Internet[1]. IPv6 represents an entirely new protocol, incorporating improved routing capability, enhanced ability to support real-time audio and video traffic, better security and privacy, as well as a much larger address space. Although IPv6 is commonly misunderstood as "IPv4 with a larger address space," IPv6 represents and entirely new protocol incorporating 4 major changes: 1) IP addresses are expanded from 4 bytes to 16 bytes, 2) the format of the packet header is simplified to include only seven fields (from 13 in IPv4) thus making routing faster, 3) various provisions are incorporated to enhance Quality of Service (QoS) and 4) security is improved through authentication and privacy capabilities. Currently, the DoD's networks, operating under IPv4, remain vulnerable to limited address space, antiquated architecture, difficulties in providing Quality of Service (QoS) for VoIP, and critical security concerns.

As the Office of Management and Budget (OMB) mandated in September 2006 the complete transfer of all DoD networks to IPv6[2] by June, 2008, the DoD is lagging behind. However, in an effort to facilitate the conversion from

IPv4 to IPv6 as mandated by the OMB, the Defense Information Systems Agency sponsored a three-phase U.S. Service Academy IPv6 Pilot Network Project.

In accordance with the OMB mandate to switch to IPv6, the Defense Information Systems Agency, developed and published the Department of Defense Internet Protocol Version 6 Generic Test Plan which provided a plan for the testing, analyzing, and validating of commercial and government IPv6 implementation throughout the DoD network. Thus, in order for a product to be considered "IPv6 capable," it must complete testing for performance and interoperability in accordance with the Generic Test Plan as well as be in conformance with the various industry-wide standards that the Internet Engineering Task Force establishes in its Requests for Comments (RFCs)[3]. This DoD IPv6 test plan "specifies test criteria and procedures for IPv6 products involved in or connecting to the Global Information Grid"[4]. The plan divides the testing of IPv6 into nine main categories: Core IP Functionality, Routing and Switching, Transition Mechanism, Common Network Applications, Security and Information Assurance, Mobility, Quality of Service, Multicasting, and Network Operations and Management. As it was assumed that the Department of Defense would be unable to transfer all of its networks, services, and information technology to IPv6, interoperability was crucial to the development and deployment of IPv6 across the DoD networks.

As a subset to the overall DoD IPv6 test plan, the Defense Information Systems Agency sponsored the U.S. Service Academy IPv6 Pilot Network Project which aims at eventually connecting the five federal service academies—the United States Naval Academy (USNA), the United States Military Academy (USMA), the United States Air Force Academy (USAFA), the United States Merchant Marine Academy, and the United States Coast Guard Academy (USCGA)—together using an IPv6/IPv4 tunnel. This process of connecting the different academies was divided into phases: phase one, currently underway, entails the initial connection of USNA with USMA; phase two will involve the connection of USAFA to this network, and phase three will see the connection of USCGA and USMMA to this network. The primary purpose of this Pilot Network Project was fourfold: 1) to "provide validation or refutation of operational concepts developed for transition from IPv4 to IPv6, to include investigation into Information Assurance, Network Management, Multicasting, Domain Name Services, and standard Transport Control Protocol (TCP) services," 2) to "develop common best practices for the utilization of IPv6," 3) to "develop IPv6 as a protocol by experimenting with its inherent capabilities for mobility, flexibility, robotics control, and convergence of services," and 4) to "provide training to the staff, faculty, and students in the next generation protocol to be used throughout the US Department of Defense"[5].

As most of the Service Academies operate on defense networks the use of tunnels between the Academies' firewalls was seen as a necessity for maintaining network integrity. With this in mind, many implementations and tests were limited by the nature and constraints of defense networks. This did, however, provide a unique perspective on the implementation capability of a different Internet Protocol with regards to information security and assurance. In addition to the inter-academy IPv6 network connection mandated by the Pilot Network Project, the scope of this research study was twofold: 1) to test and develop convergence techniques for the coexistence of IPv4/IPv6, and 2) to discover and analyze the ramifications that the transition to IPv6 would have on legacy systems. In addition, we found that it is vital to understand the IPv6 addressing scheme as it provides the critical and fundamental underpinning to the many other changes made to the new protocol. To help facilitate the conversion to IPv6, we have built, tested, operated and maintained a pilot IPv6 network between the United States Naval Academy in Annapolis, Maryland and the United States Military Academy in West Point, New York. This network hosted multiple operating systems and employed a Domain Name System server, a web server, a File Transfer Protocol server, a Dynamic Host Configuration Protocol server, in addition to multiple PC and Unix-based clients. This network was then used to provide validation and refutation of operational concepts developed for the transition from IPv4 to IPv6. This research included investigations of network management, address allocation, Domain Name Services, protocol capabilities and communication, as well as transition techniques for migrating from the IPv4 protocol to IPv6. This research intended to serve as an integral part of the first phase in the DISA Pilot Network Project which entailed the building, operating and maintaining of a pilot IPv6 network between the United States Naval Academy, and the United States Military Academy and was then to be used to provide both validation as well as refutation of operation concepts and best practices for the transition from IPv4 to IPv6, including investigations of network management, address allocation, and Domain Name Services, in addition to the testing and development of transition techniques for migrating from the IPv4 protocol to the IPv6 protocol. We present the results of our experience in deploying and testing an IPv6 network, discuss proposed best practices for the utilization of IPv6, and provide a list of lessons learned to include various incompatibility issues that were noted with many popular software applications.

2. IPv6 Addressing

IPv6 differs in many significant ways from IPv4; the first and foremost difference exists in the addressing mechanism that is utilized. One of the many issues raised with the IPv4 protocol was the 32-bit address structure; which effectively limited unique available addresses to 4,294,967,296. IPv6, in response to the growing exhaustion of IPv4 address space, provides a 128-bit long address, thus yielding 2^{128} unique IPv6 addresses. In order to shorten the notation of IPv6 addresses, each address is represented in hexadecimal rather than decimal notation.

The IPv6 addressing architecture classifies addresses as one of three types: unicast, multicast, and anycast[6]. A unicast address is an address that identifies a single node. Unicast addresses are further divided into three subtypes: link-local, site-local, and global[7]. Additionally, IPv6 unicast addresses are derived from the MAC address of the host node as can be seen in Figure 1.

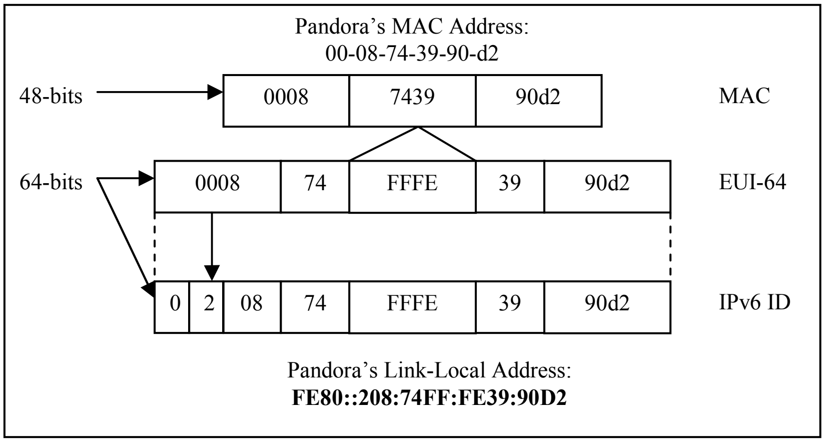

Figure 1. IPv6 unicast addressing derivation

Multicast addresses, as the name implies, are addresses that identifies groups of nodes[8]. This serves as a replacement to the IPv4 broadcast addresses and identifies a group of nodes so that a packet with a multicast address is sent to all those belonging to the multicast group. This group is normally found on a given site. The last major addressing scheme change in IPv6 is the anycast address. According to RFC 3513, the anycast address is a unicast address assigned to multiple machines[9]. Any packet sent to an anycast address is delivered to the nearest available interface configured for it. However, as the anycast address is virtually indistinguishable from the unicast address, the nodes must be configured for that address.

In summary, IPv6 provides a new addressing scheme meant to address the shortcomings and failures of its predecessor, IPv4. Although the addressing change may not be the most important or biggest change in IPv6, it is probably the most noticeable and necessarily impacted the course of this research project.

3. Research Study

As this part of the research was dedicated to the transition from IPv4 to IPv6, the establishment of a basic IPv6 network was seen as primary; therefore, fundamental to its inception were the use of basic and commonly used software and operating systems. Thus, rather than connecting directly to the United States Military Academy or diving into the configuration of routers, we decided on developing the network in an incremental method by first starting with a simple three computer and one hub configuration as a pragmatic foundation on which to build. Due to its wide use and acceptance by both the DoD and the general public, the operating system (OS) chosen for this initial configuration was Windows XP SP2. Our overall network design is illustrated in Figure 2.

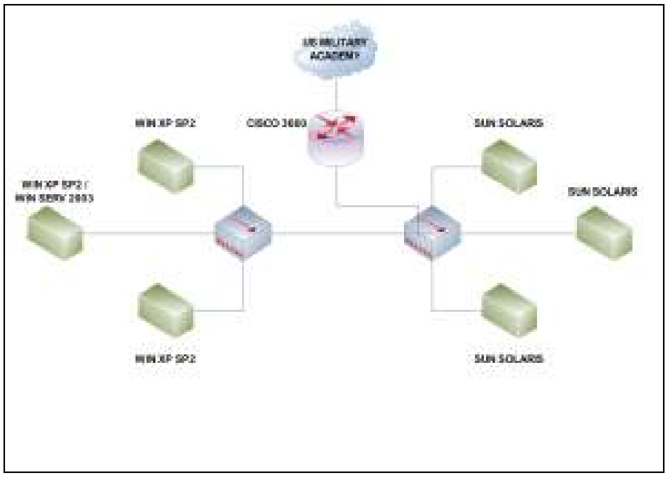

Figure 2. Overall IPv6 network setup

3.1 operating system compatibility and basic connectivity

Before beginning with the installation and setup of an IPv6 network, we first decided to test for basic IPv6 compatibility with Windows XP SP2. It was discovered after skimming the Microsoft support web pages that IPv6 is preinstalled as a package on all Windows XP SP2 systems – though it is not initially setup. This is in stark contrast to Solaris 10, SUSE 10.1 and other Linux flavors which, by default, have IPv6 enabled. However, on Windows XP SP2, a user can setup IPv6, thus assigning link-local addresses to their node, by using the netsh command-line utility. The netsh tool also provides the ability to configure a router on a Windows XP system with forwarding and advertising enabled, to create static routes, and to assign site-local IPv6 addresses and address schemes.

After installing and configuring the IPv6 package on the three computers, we connected them to a 4-port hub. We successfully pinged (ICMP echo request/echo reply exchanges) all the computers using the IPv6 link-local addresses. Though IPv6 literature references a ping6 that must be used when pinging IPv6 address, Windows XP SP2 syntactically determines what type of address you are pinging when you use the normal ping command. Thus, the Windows XP SP2 ping function is IPv4/IPv6 interoperable.

However, in order to ensure that the traffic over the hub was actually IPv6, and not IPv6 wrapped in IPv4, or just IPv4, we installed Wireshark on the three computers. As Wireshark is a protocol analyzer that allows the user to examine in detail the raw traffic being placed on a network with the added functionality that Wireshark incorporates the filter "ipv6" to filter IPv6 traffic, we were able to confirm that IPv6 link-local addresses were indeed being used for neighbor advertisements, neighbor solicitations, and pings; this traffic was also marked as ICMPv6 in the Wireshark display.

3.2 telnet and file transfer protocol

Having established and tested the basic and initial setup, we decided to test the basic services so that we could empirically observe and validate data communication and transfer other than ICMPv6. We first began testing telnet terminal emulation services. After setting up basic users and logins and opening port 23 in the Windows Firewall, we were able to create a successful telnet connection between two machines using standard telnet commands and IPv6 link-local addresses. The connection could be further validated as IPv6 by referencing the output of the netstat command.

As we had successfully connected via telnet, we decided to test the File Transport Protocol (FTP). This, however, was initially unsuccessful. File Transport Protocol, on the Windows XP SP2 machine, is a member of the Microsoft's Internet Information Services (IIS) suite. These services include FTP, Simple Mail Transfer Protocol (SMTP), Network News Transfer Protocol (NNTP) and web. As of IIS 6.0, the EnableReverseDnsLookup, which is used by IIS 6.0 to perform reverse Domain Name Service (DNS) lookups for names of client computers, is not IPv6 compatible. This is the primary reason FTP does not work natively on Windows XP using IPv6[10]. According to Microsoft's technical support pages, SMTP and NNTP are also not supported under IPv6[11].

Having determined that IIS 6.0's FTP service cannot host an FTP connection using IPv6, it remained to be determined whether there was an IPv6 FTP client/server program freely and readily available. To determine if this incompatibility could be abrogated by using third-party software, we installed and utilized XLight's FTP Server program. This program was capable of successfully hosting FTP connections using IPv6 only. Ergo, though FTP under IPv6 is not natively supported with Windows XP, it is still possible with other vendor FTP programs.

3.3 expanding the IPv6 network

After testing basic connectivity between three Windows XP SP2 machines using IPv6, we decided to test UNIX compatibility with IPv6 and interoperability between Windows XP SP2 and UNIX utilizing IPv6. Thus, we installed Sun Solaris 10 on three other machines and connected these machines to another hub. The two hubs were then connected effectively connecting the three machines running Windows XP and the three machines running Sun Solaris 10. Sun Solaris 10, as well as many other UNIX and Linux based operating systems, are installed with IPv6 support setup and, in most cases, running. As we were still only testing basic connectivity, the lab was set up with only link-local addresses. After connecting the hubs, we were able to successfully ping between the Sun machines and the Windows machines. We were then able to successfully telnet between the machines. In addition, FTP is natively supported by Sun Solaris 10 using IPv6.

As we had completed many of the basic data communication and transfer tests, we decided to install Windows Server 2003 SP1 Enterprise Edition on one of the Windows Machines to test FTP, DNS, DHCP and Web Server services when running on Windows Server 2003. However, many of the services of Windows Server 2003 run on IIS 6.0. Ergo, the creation of an FTP server resulted in failure for the same reason as above.

3.4 establishing a Domain Name Service (DNS) Server

After testing and determining the incompatibility of IPv6 with the FTP server, we determined to test setting up a Domain Name Service (DNS) server. However, due to the complexity involved, we decided to use Microsoft's *Step-by-Step Guide for Setting Up IPv6 in a Test Lab*[12]. This publication initially suggests setting up a network with three segments utilizing two systems as routers. Because the machines in the lab had only one Ethernet port and one Network Interface Card (NIC), and in order to simplify the setup, only one segment was created. This configuration was also chosen as it most closely represented our current setup with the two hubs and six machines.

We quickly learned something very important: Microsoft's guide does not create an IPv6-only network, but rather uses IPv4 for the DNS setup. Later, in fact, it was determined that Microsoft's DNS on Windows Server 2003 does not support DNS solely using IPv6. Realizing that Microsoft was not quite ready for IPv6, we nevertheless, after following the guide, created a network that provided IPv4/IPv6 capabilities. After setting up the individual IP addresses, a forward lookup zone was created.

As there were no other routers on the test network, there was no need to add a next hop to the static route. However, in the case of using one DNS over a tunnel or using multiple virtual local area networks (VLANs), one would most likely setup a static route with multiple hops. This setup provided the network with a site-local address scheme. It is of importance to note that until this point, the use of link-local addresses is more than sufficient for the maintenance of a network. However, after implementing certain network services often considered crucial to modern day networks, the assignment of a site-local addressing scheme to the link is necessary. As was the situation with the link-local addresses, the Extended Unique Identifier (EUI-64) based interface ID was again appended onto the end of the site-local address scheme. After the network was setup, AAAA DNS records (IPv6 DNS records) were created for each host (the site-local address was used). As Sun Solaris 10 lacks much of a graphical user interface (GUI) for the assignment of gateways and a DNS, several files needed to be extensively modified in Sun Solaris 10.

After setting up the DNS and creating the IPv6 DNS records for each host, we tested the DNS by pinging from each computer using the computer name rather than the computer's site-local or link-local address. After the testing was successful, we decided to test a telnet session using the DNS host name which was again successful. However, nslookups were not feasible as a reverse lookup zone was not created. After creating a reverse lookup zone and inserting the DNS host name records for IPv6, nslookups on client computers were successful. Because DNS was functional and had tested successfully with IPv6, we decided to set up a Web Server. We created a basic webpage.

After setting up the web server, we attempted to connect to the website using Mozilla's Firefox. Firefox successfully connected to the web server if we placed the IPv6 address in brackets. The reason for the brackets around the IPv6 address is to prevent a parsing of the IPv6 address and the misinterpretation of the colons as the port to which to connect. Firefox also successfully connected using DNS names and bypassing the IPv6 address. Internet Explorer v6.0, however, was not IPv6 compatible with our setup. We attempted both the use of the IPv6 address in brackets as well as the use of the DNS name – both were unsuccessful.

3.5 connecting the IPv6 network to the US Military Academy

When we had decided to connect to the established tunnel with the US Military Academy, we began by trying to establish a clear IPv4 connection. However, due to DoD network security limitations, the Access-Control Lists (ACLs) and Firewalls between USNA and the US Military Academy, prevented us from establishing an IPv4 connection and from troubleshooting the connection problem. After corresponding with the Military Academy, it was decided to forego the IPv4 connection and attempt an IPv6 connection, as the Military Academy's ACLs and firewalls were configured to allow IPv6 traffic through. This, however, brought only marginal success as the packet success rate was 46%. What we noticed was that every other packet was being dropped by the Military Academy. Again, concerned that the ACLs or the firewalls were preventing a clear connection, we contacted the Military Academy. However, it was determined that the Military Academy had assigned their IPv6 address to an actual interface instead of using the Loopback 6 address. Because the traffic had to be routed internally by their router before being sent back, the ping program was timing out before receiving the reply. However, after reconfiguring their router, we were able to get 100% success rates between us and them. After deleting the route and site-local address scheme that was configured previously and connecting the network to the CISCO 3600 router that was attached to the tunnel, the network took the site-local scheme assigned by the router. After each machine updated its site-local address, all network services previously set up worked as before.

After having achieved the goal of the first phase of the DISA Pilot Network Project, we decided to attempt to setup the DNS as IPv6 only. However, this attempt failed, most likely due to static routing issues. This necessitated the creation of static routes for the CISCO 3600 router. Towards the end of the research study, we desired to be able to test the bandwidth and data transfer rates and times. Because this would necessitate the synchronizing of the computer clocks, we attempted to set up a Network Time Protocol (NTP) server on the network so that we could sync all the machines. However, the NTP service as provided by Windows is not IPv6 supported. Furthermore, because we lacked other testing hardware or software that would allow for us to test latency and bandwidth, we were unable to test for differences between IPv4 and IPv6 data transfer and throughput on the network.

3.6 summary

In summary, we were able to take a simple, three-node network and setup a basic functioning IPv6 network using Windows XP SP2. After testing, we successfully setup telnet and FTP (using third party software) connections and verified the traffic was indeed IPv6 via Wireshark. This network was developed further by the addition of three machines running a Sun Solaris 10. By adding Windows Server 2003 to one of the machines, we were able to establish a DNS that functioned using both IPv4 and IPv6. We then connected this network to the CISCO 3660 router and established a working connection with the United States Military Academy. The following table provides a basic summary of our tests and results.

Table 1. summary of tests and results achieved

Service Tested	IPv6-only	IPv6 with IPv4	WIN XP SP2	SUN SOLARIS	Other Software
Ping	Y	N	Y	Y	N/A
Telnet	Y	N	Y	Y	N/A
FTP	Y	N	N	Y	Y
DNS	N	Y	Y	Y	N/A
NTP	N	N	N	N	N/A
DHCP	N	N	N	N	N/A
Active Directory	N	N	N	N	N/A
SNTP	N	N	N	N	N/A
IIS 6.0	N	N-IPv6/Y-IPv6/IPv4	N-IPv6/Y-IPv6/IPv4	N	N/A
IExplorer v6.0	N	Y	Y	N	N/A
Mozilla Firefox	Y	Y	Y	Y	N/A

4. VoIP and Ongoing Research

Currently, we are researching the application of Voice over Internet Protocol (VoIP), QoS, and the interoperability implications with IPv6 over military networks. As stated previously, one of the main problems with implementing VoIP is the fact that the main protocol promoted and backed by the civilian sector is Session Initiation Protocol (SIP). SIP is fundamentally a peer-to-peer protocol, thus making it unsuitable for implementation across DoD networks. However, as SIP is being pushed by many industry leaders, the DoD is now faced with adapting its network structure and security to be compatible with these new protocols, rewriting and developing its own protocols, or not implementing a technology that would exceedingly beneficial for military applications.

5. Results

In our research, we were able to build a fully functioning network that implemented IPv6. In addition, we were able to successfully test and validate the compatibility of Windows XP Professional SP2 and Sun Solaris 10 with IPv6. However, with our setup, only basic network functionality was possible using IPv6 only. Only through the implementation of an IPv4/IPv6 network were we able to implement other services – especially those requiring a Domain Name Service that allowed for both forward and reverse lookups; allowing nslookups, pings, file transfers, and telnet services based on names rather than IPv6 addresses. After the creation of the network and the establishment of common network services, we were able to connect the network to an IPv4/IPv6 VPN tunnel; thus completing the primary phase of DISA's Pilot Network Project. In addition, we were able to validate the address changes of IPv6 and were successful in the manipulation of those addresses to different address schemes.

We determined that FTP as implemented by Microsoft's IIS 6.0 is not IPv6 compatible. As this is predicated upon the incompatibility of EnableReverseDnsLookup with IPv6, many other services supported by IIS 6.0 are also incompatible: DHCP, Active Directory, SNMP, NNTP, and NTP. However, DNS, as per our testing, is compatible if the network is an IPv4/IPv6 network. In addition, our findings indicate that for the web server, Internet Explorer v6.0 is also not compatible with IPv6.

6. Conclusion

The goals we established from the onset of the research study were twofold: 1) test and develop IPv4/IPv6 convergence techniques involving a fully-functioning IPv4 network, and 2) test and develop inter-protocol communication and transition techniques specifically including legacy systems. In addition, it was an underlying goal to partake in and accomplish the initial phase of DISA's Pilot Network Project.

In an effort to test and develop IPv4/IPv6 convergence techniques, many of the common services expected of a network were used as a baseline for testing. Many of the services were, however, limited more by the use of common, mainstream operating systems such as Windows Server 2003. It has been documented, though not fully tested, that DNS BIND has been successful as an IPv6-only DNS[13]. Other documentation has suggested the existence of NTP and SNMP services in an IPv6-only environment. These documents are important as they demonstrate the current viability for IPv6-only networks that provide the same services as common IPv4 networks. However, the current latest version of Windows Server limits the usability of many of the services to strictly an IPv4/IPv6 network. To be sure, while the eventual goal is that IPv6 replace IPv4, coexistence of the two protocols must be allowed as IPv4 will be around for long time yet. What the implementer must be cautious of is that applications, software, and operating systems do not become the limiting reagent for the process of change. As the research study has shown, Windows Server 2003 could be such a limiting reagent if relied upon.

The second goal was not entirely met. This was largely due to the difficulties in setting up an IPv6 network with common, up-to-date systems. However, the fact that the implementation of IPv6 was difficult on today's systems gives a hint at the possible difficulties in attempting to implement IPv6 on legacy systems. One area that shows promise, however, is the use and implementation of Linux and Unix-based operating systems. Many Linux and Unix based systems already have IPv6 enabled. Others have the capability, though it is disabled by default. Also, as Linux and Unix-based systems tend to be more easily updateable and patchable, the limit to these systems is more concentrated with the physical hardware than with the capability of the OS.

Another fundamental aspect of the research was the impact that necessary DoD security limitations had on our research. Many of the connectivity issues we had with the Military Academy were due entirely to access control lists, firewalls, and router configurations. Thus, while some of this research and testing would be relatively simple in the civilian world, many aspects became rather complex with the added impositions of DoD network security.

In conclusion, IPv6 is a new protocol that provides many fundamental changes to the widely standardized and implemented IPv4 protocol. As shown previously, the global need for addresses and a better protocol demands the implementation of IPv6. However, the implementation of this new protocol is limited largely by the software and systems widely used and the alacrity with which major software vendors pursue the ability to seamlessly implement IPv6 in a largely IPv4 world.

7. References

[1] V. Cerf, "On the Evolution of Internet Technologies", *Proceedings of the IEEE, Vol. 92, No. 9* (IEEE, 2004), 1363.

[2] Executive Office of the President, Office of Management and Budget, "Transition Planning for Internet Protocol Version 6 (IPv6)," The White House, http://www.whitehouse.gov/omb/memoranda/fy2005/m05-22.pdf, 2.

[3] "Department of Defense Internet Protocol Version 6 Generic Test Plan Version 2 September 2006," The Joint Interoperability Test Command, http://jitc.fhu.disa.mil/adv_ip/register/docs/ipv6_gtp_v2.pdf, 5.

[4] Ibid., 5.

[5] "U.S. Service Academy IPv6 Pilot Network Project", DISA Project Proposal, .

[6] C. Popoviciu, E. Levy-Abegnoli, and P. Grossetete, *Deploying IPv6 Networks* (Cisco Press, 2006), 27.

[7] Ibid., 27.

[8] Ibid., 40.

[9] Ibid., 39.

[10] "How IIS 6.0 Supports IPv6 (IIS 6.0)," Microsoft TechNet, http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/1ecff3af-36c2-41b5-957a-8bcc6fac8abc.mspx?mfr=true.

[11] Ibid.

[12] "Step-by-Step Guide for Setting Up IPv6 in a Test Lab," Microsoft TechNet, http://technet2.microsoft.com/windowsserver/en/library/0af83b7a-16d7-4c8e-856b-22b7b65ceb7b1033.mspx?mfr=true.

[13] D. Koren, "Are we ready for IPv6? Is IPv6 ready for us?" (International Journal of Network Management, 2005), 62-63.