2014

University of North Carolina Wilmington

Master of Science in

Computer Science and Information Systems

Proceedings

https://csbapp.uncw.edu/mscsis

AN APPLIED CASE STUDY OF AN IPV6 CONVERSION IN A SIMULATED MIDSIZED BUSINESS

Michael Abate

A Capstone Project Submitted to the
University of North Carolina Wilmington in Partial Fulfillment
of the Requirements for the Degree of
Master of Science in Computer Science and Information Systems

Department of Computer Science
Department of Information Systems and Operations Management

University of North Carolina Wilmington

2014

Approved by

Advisory Committee

_____	_____
Dr. Ron Vetter	Dr. Jeff Cummings

_____
Dr. Douglas Kline, Chair

Abstract

An Applied Case Study of an IPv6 Conversion in a Simulated Midsized Business. Abate, Michael, 2014. Capstone Paper, University of North Carolina Wilmington

The purpose of this research is to address the eminent exhaustion of IPv4 address space by gaining knowledge and practical experience of the replacement protocol, IPv6 by performing a conversion of an IPv4 SMB network to IPv6. IPv4 has been the foundation of the Internet for the past three decades, but it is a victim of its own success, and we are faced with the issue of running out of IPv4 addresses to hand out. This research reviews the IPv4 and IPv6 protocols including a comparison of the two. Following this review, the paper then examines the mechanisms available for transitioning from IPv4 to IPv6. An IPv4 network was designed and built to simulate a mid-sized business. The network then underwent a conversion to IPv6 to understand the potential issues and impact of transitioning to the new protocol. The final results provide a case study for networking professional considering a migration to IPv6. A suggested project plan is included as well as a number of the common issues encountered during the conversion process.

Table of Contents

List of Figures

List of Tables

List of Diagrams

Chapter 1: Introduction

Internet Protocol (IP) is the standard that computers follow in order to communicate with each other. IP can be compared to the telephone numbering system because the IP system allows computers to exchange information by using a unique identifier, like a phone number. Phone numbers, as a unique identifier, are standardized by using the same amount of digits in each number. Internet Protocol version 4 (IPv4) sets the unique identifier for an IP system, known as the IP address, to have a length of 32 bits. IPv4 has been the foundation that sets the standards for Internet communications for over 3 decades. It was originally designed in the 70's with the premise that Internet communication would remain within governmental and educational domains. Fast forward more than 30 years, Internet communications and devices that communicate using the Internet have grown so much that the Internet is now presented with the dilemma of running out of IP addresses. American Registry of Internet Numbers (ARIN), the authority of IP addresses for the Americas, reports that as of September 3, 2010 they have allocated 94.5% of the IP Pool [1]. Because the founders did not anticipate the huge growth of the internet when they designed the IPv4 protocol, the decision was made to go with a smaller addressing scheme.

Surprisingly, at the time when IP was being developed, designers actually considered making the addresses 128 bit. Vint Cerf, known as a co-founder of the Internet, had the final say when making the decision on whether to go with a 128 bit address space or 32 bit [2]. When Cert was interviewed by the Business of Federal Technology, Cerf recalled:

> Some researchers wanted a 128-bit space for the binary IP address…But others said, "That's crazy," because it's far larger than necessary, and they suggested a much smaller space. Cerf finally settled on a 32-bit space that was incorporated into IPv4 and provided

> a respectable 4.3 billion separate addresses. "It's enough to do an experiment," he said. "The problem is the experiment never ended." [3]

The designers, including Cerf, never imagined the Internet would grow to the size it is today, with over 4 billion addresses. It is widely known that IPv4 will run out of IPs. The following Figure 1 provides a percentage of IPs remaining by region.

Figure 1. IPv4 Address Report

IPv4 Address Report

This report generated at 26-Oct-2013 09:11 UTC.

IANA Unallocated Address Pool Exhaustion:
03-Feb-2011

Projected RIR Address Pool Exhaustion Dates:

RIR	Projected Exhaustion Date	Remaining Addresses in RIR Pool (/8s)
APNIC:	**19-Apr-2011** (actual)	0.8265
RIPE NCC:	**14-Sep-2012** (actual)	0.8554
ARIN:	**17-Jan-2015**	1.6561
LACNIC:	**26-Mar-2015**	1.7469
AFRINIC:	**31-Jul-2022**	3.5422

Internet Protocol version 6 (IPv6) was primarily developed to deal with the limited number of IP addresses of IPv4 by using a 128 bit address [4]. It is projected as seen in Figure 1 that the Americas will run out of IP by the 17th of January 2015. In order to continue the growth and success of the Internet and to avoid running out of IPs as well as avoid Internet interruption(s), migration from IPv4 to the IPv6 standard must occur.

In order to make up for the lack of IP addresses, over the years IPv4 has had many patchwork solutions added to the protocol, many of which affect the functionality. One of which is Network Address Translation (NAT), the process of converting the IP address from one address to another, i.e. translating from a public IP address to a private address [5]. NATing was

introduced to resolve the IP address shortage issue of IPv4. NATing allows for one IP address to cover many IP addresses of a different network, by introducing a device between networks that handles the translation between networks. One of the problems with the need to use address translation is that it compromises the security and integrity of the communications. IPv6 addresses NATing issues by providing enough IP addresses so that there is no longer a need to "hide" multiple IP addresses behind a single point IP. Yet, despite the benefits of IPv6, there seems to be a hesitation among corporations and Internet Service Providers (ISP) to adopt IPv6. This hesitation is most often due to the lack of understanding of the options and benefits associated with the converting from IPv4 to IPv6 [6].

The goal of the current research was to build an IPv6 migration case study while addressing the challenges of moving an IPv4 network to the new protocol. The aim of this research was to (1) gain an understanding and familiarity of the IPv6 protocol, (2) understand why it is imperative that the Internet community phase out IPv4, and (3) provide a case study on how to migrate an existing IPv4 system to an IPv6 system. A thorough understanding of the protocol will help support the push to replace IPv4 by defining IPv4 shortcomings, the similarities and differences between IPv4 and IPv6, as well as the inherit benefits of IPv6. Completing these objectives will bring awareness to the criticality of the need to replace IPv4 as well as describe a viable option to a conversion.

To achieve these goals this research simulated an existing small to medium size IPv4 corporate network and document the migration to IPv6. The experiment included a stage of coexistence via a technology known as dual stack. This project was limited to a Local Area Network (LAN) in order to avoid ISP collaboration. A LAN can be managed by a sole entity, such as a small to mid-sized business, but an ISP would have introduced time delays due to

policies, and red tape. Because time was a pivotal constraint, introducing an ISP was not practical nor was there a need to include the added complexity of working with an ISP. Small to mid-sized businesses (SMB) include resources such AD, servers, laptops, desktops, printers, network switches, DNS, and DHCP services. The components of a SMB needed to be converted from IPv4 to IPv6. Confining this project to a LAN protected the research from uncontrollable factors that would have introduced the need to use other migration techniques, including tunneling and NAT. In the following chapter, a review of the current literature concerning IPv4 is discussed as well as the benefits associated with the conversion to IPv6.

Chapter 2: Review and Analysis of Literature

2.1: Introduction

Before embarking into the conversion, it was important to understand both protocols. The following sections research key areas of both protocols, including the similarities and differences, as well as the enhancement that IPv6 offers. The review begins by examining the Datagram and will move up the Transmission Control Protocol/Internet Protocol Model.

2.2: Datagram

In 1981, John Postel submitted the Request for Comment number 791 (RFC 791) titled "Internet Protocol". The document specified the standards to be used by government entities, and would later be referenced as IPv4. The RFC's motivation was stated as:

> "The Internet Protocol is designed for use in interconnected systems of packet-switched computer communication networks. The internet protocol provides for transmitting blocks of data called datagrams from sources to destinations, where sources and destinations are hosts identified by fixed length addresses. The internet protocol also provides for fragmentation and reassembly of long datagrams, if necessary, for transmission through "small packet" networks." [7]

The addresses he refers to are called Internet Protocol Addresses and are discussed later in the research. Postel also mentioned the *datagram*. A datagram is defined as a self-contained, independent entity of data carrying sufficient information to be routed from the source to the destination computer without reliance on earlier exchanges between this source and destination computer and the transporting network [8]. To elaborate on this definition, datagrams are structured chunks of data consisting of a header and the data, known as the payload. The

header's responsibility is to carry the vital information for the proper handling of the datagram and includes information such as source IP, destination IP, fragmentation instructions, and version. The header information is only used as communication instructions. The data that will be shared with the receiving computer is contained in the payload. Today, the Internet transmits the majority of the data using the Internet Protocol (IP). The data being transferred through IP communications is broken up into datagrams. RFC 760 was the first document to refer to IPv4 protocol but was later superseded by RFC 791.

Almost 20 years later in December of 1998, a new RFC was published (RFC 2460) titled "Internet Protocol, Version 6 (IPv6)". IPv6 has been developed based on the rich experiences we have from the developing and using of IPv4. Proven and established mechanisms have been retained, known limitations have been discarded, and scalability and flexibility have been extended [9]. Table 1 compares the similarities and differences of the IPv4 and IPv6 Headers. The white lines are fields of IPv4, the items highlighted in orange are fields that are present in both protocols, and the fields highlighted in green are fields present in IPv6 headers. The most notable differences are the version field, no checksum, IP header length, and Path MTU discovery [7] [4].

Table 1. Datagram Table comparing IPv4 and IPv6

	IPv4	IPv6	Comments
Version	4 bit; 4	4 bit; 6	
Traffic Class		8 bits;	IPv4 Type of Service field
Flow Label		20 bits; QoS	Compared to Type of Service
Internet Header Length	4 bits	Not present reducing processing	
Type of Service	8 bits; QoS		
Total Length	16 bits;		
Payload Length		16 bits	

Identification	16 bits; ID of Fragment		
Next Header		8 bits; IPv4 Protocol field	IPv4 Protocol field
Flags	3 bits; Fragment Status		
Fragmentation Offset	13 bits;		
Time to Live	8 bits; Header Processing amount		
Hop Limit		8 bits; Count decremented by 1 each time datagram forwarded	IPv4 TTL field
Protocol	8 bits; Next protocol		
Header Checksum	16 bits; Verified at each instance the header is processed		Checksum removed due to the presence of a checksum in upper layers
Source Address	32 bits	128 bits	
Destination Address	32 bits	128 bits	
Options	Variable		
Padding			

The following fields were chosen to examine further because they represent the most significant change between the protocols:

- Version field – this field is straight-forward and indicates IPv4 or IPv6. This field informs the receiver of the datagram on how to process the datagram.
- Checksum – The checksum is the process of confirming the header did not change during the trip between source and destination. In IPv6, the Header Checksum was removed to improve processing speed. At the time when IPv4 was developed, check summing at the other layers was not as common, so checksum field in the IPv4 header made sense. Today the risk of undetected errors is minimal [9].

- Header Length - IPv4's header has a variable length. As stated in RFC 2460, the IPv6 Header has a fixed length of 40 bytes. IPv6 header is therefore much simpler and leaner then the IPv4 header. [9] Having a variable such as IPv4's header introduces an unknown that requires a process to determine. Because IPv6 has a fixed header length there is no need to have a header length field. Setting the header to a fixed length allows the elimination of extra steps and, again, makes the processes more efficient. The slight yet imperative changes to the header have simplified header of IPv6 proves to be more efficient.

2.3: Benefit of IPv6

One of the reasons the Internet has been successful is the ability interoperate with a variety of technologies and communication lines. When two devices communicate, it is highly unlikely that they are communicating directly with one another. In the majority of cases, there are many devices in between the two systems trying to communicate. These intermediary devices are the brains of what transitions the communications from one technology to another. One of the steps of these intermediary devices, sometimes known as a router, is to modify the size of the datagram to make it conform to the size needed for the next section of the datagram's path. Each router may be connected to many different datagram sized capable communication technologies.

2.3.1: Maximum Transmission Units

With IPv4 the protocol allowed each router to determine the size of the datagram. The router would process the datagram by dividing it up into more manageable units. This process is known as fragmenting. When the router determines the maximum size a datagram can be this is known as the Maximum Transmission Unit (MTU) [7]. The process of determining the size

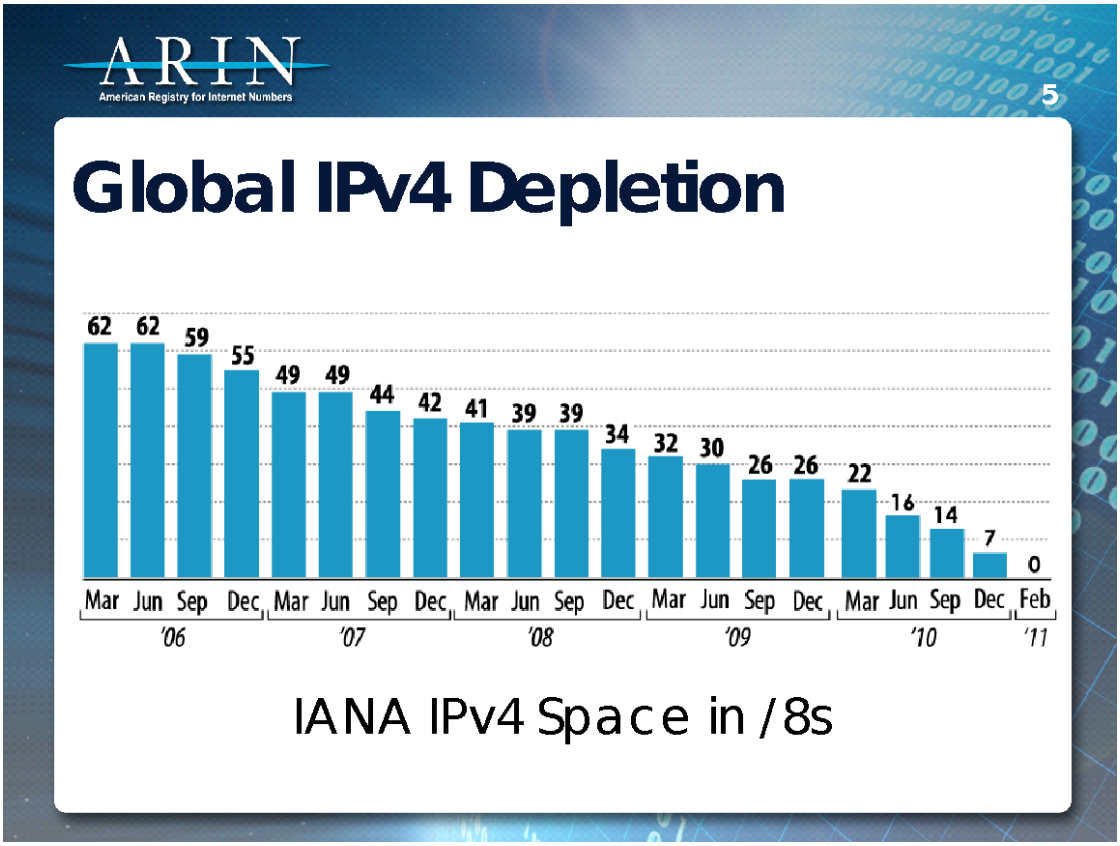
needed for the datagram's path is known as the Path MTU Discovery [10]. As stated in RFC 1191, IPv4 had the process of Path MTU Discovery but was not utilized due to the routers doing the next path MTU determination.

IPv6 moves the fragmentation processes from the intermediary devices to the end nodes. The source performs a Path MTU discovery that determines that maximum size the path is capable of and then breaks the datagram into chunks of that size [11]. This method is more efficient because of two reasons, the routers no longer perform the fragmentation, reducing the processing, and the process only occurs once, which is at the source of the path. Luckie and Stasiewicz's research provides data that reinforces the comments that Path MTU is not utilized in IPv4 because the majority of Path MTUs on IPv4 networks are blocked by firewalls. Their research did see an increase in Path MTU successes from IPv4 to IPv6 [12]. The reason for this result is that IPv6 requires the Path MTU discovery to be successful in order for a MTU size to be properly sized.

2.4: Addressing

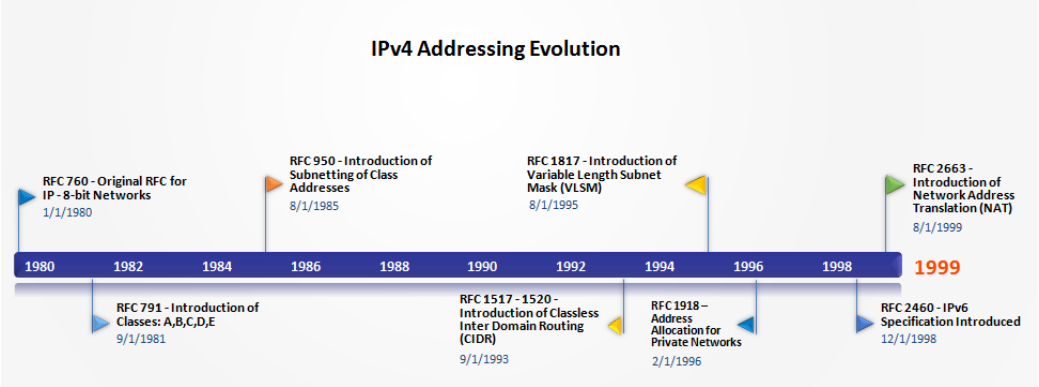
IP Addresses are the unique identifiers of hosts, devices and systems connected to an IP network. An IP is used to communicate with another device on the network. Without IP addresses, computers would be unable to communicate with each other. IPv4 protocol was specified to have has approximately 4.3 billion IP Addresses [7]. According to the American Registry for Internet Numbers (ARIN), the remaining 5 Class A addresses were assigned in February of 2011 [13]. Over the past decade, the rate at which the available IPs has been depleting has been on the rise.

Figure 2. Global IPv4 Depletion



The reason for the rapid increase in the depletion of the IPv4 addresses is the ubiquity of devices capable of accessing the Internet [14]. Today, it is not uncommon for an electronic device to have the capability to access the Internet. Although over the last decade, the depletion rate has increased, it has been known by scientist that the IPv4 addressing was inadequate since its inception in the early 80s. As the following timeline depicts, it is obvious that efforts to prolong the exhaustion were started at very early stages of the IPv4 protocol. See Figure 3.

Figure 3. IPv4 Addressing Evolution



Some of the most significant attempts at prolonging the inevitable exhaustion of IPv4 Addresses are illustrated on the timeline. Immediately following RFC 760, the original RFC for the IP protocol, it was apparent that the addressing methods were inefficient. The first attempt at repairing the protocol was described in RFC 791, where the specification of Address Classes was introduced. RFC 791 specified that the IPv4 addresses be divided into five categories; only three classes would be assigned for IP networks [7].

Table 2. IP Classes

Class A 1.0.0.0 – 127.255.255.255	Addresses were intended for very large entities like governments. A maximum of 16 (2^{24} = 16,777,216) hosts and 128 (2^{7}) networks.
Class B 128.0.0.0 – 191.255.255.255	Addresses were intended for large organizations like universities. Class B can support (2^{16} = 65,536) host and 16,384 (2^{14}) networks.
Class C 192.0.0.0-223.255.255.255	Addresses were intended for smaller organizations. Class C can support 256 (2^{8}) hosts addresses and 2,097,152 (2^{21}) networks.
Class D (multicast) 224.0.0.0-239.255.255.255	Multicast addresses have a maximum of 268,435,456 (2^{28}) addresses
Class E 240.0.0.0 – 247.255.255.255	Addresses space is currently reserved

A few years after RFC 791, RFC 950 was published. Which was the implementation of subnetting. Subnetting specified the division of a classed network for the assignment to smaller entities. This mechanism allowed for the network ID to be broken into bits other than 8, 16, or

24; such as network IDs like 9 or 28 bits. This was advantageous because it introduced network size allocations much more appropriate for its applications. Businesses with only a few computers requesting an Internet presence were able to obtain a more appropriately sized block of addresses, significantly reducing the waste.

In the mid-90s the Internet Engineering Task Force (IETF) introduced classless inter-domain routing (CIDR) and variable length subnet masking (VLSM). Both CIDR (RFC 1517) and VLSM (RFC 1817) were specifically designed to impede the rapid exhaustion of IPv4 addresses.

Up to 1996, all devices were part of the same public IP pool, even though many did not need to communicate with other devices that were not part of the same domain. In February 1996, RFC 1918, (Address Allocation for Private Internets), set aside IP ranges that were not routed over the public Internet, and private entities had to configure their own devices to route these specific IPs to and from internal broadcast domains. At this point, the Internet IP address authority added extra levels of scrutiny to the justification of obtaining an Internet IP. If the device needing an IP did not need to communicate with a host of another enterprise, it was not an outside service like email, or ftp, or was not responsible for layer 3 communications; the device did not get an Internet IP [15].

By 1998, engineers started to write specifications to replace IPv4, and in December of 1998, RFC 2460 was submitted. By this time the Internet was already past a point that the standard protocol could be replaced overnight and solutions to the delay the increasing issue of IP depletion had to continue.

The time came that another stop gap solution needed to be implemented to assist in the continued popularity and growth of the Internet. In September of 1999, Network Address

Translation (NAT) was introduced in RFC 2663. IPv4 NAT was the process of translating one IP address to another. This allowed a private entity to obtain a minimal amount of Internet IPs to still interact with the Internet. This was achieved by combining RFC 1918's private IP Addresses with NAT. An entity would obtain the minimal needed IPs to interact with the Internet, yet the private section of the network would use a private IP Address scheme. If a device on the private network needed to access the Internet it passed the gateway and a device translated the IP from private to the public. This solution proved to be advantageous, but also introduced some issues. IPv4 NATing provided a mechanism to prolong the depletion of IPs, but also broke the true end-to-end communication because an intermediary device must modify the destination or source IP address. According to the Center for Next Generation Internet, 70% of the Fortune 1000 companies use NAT technologies [16].

All of these band aid fixes served their purposes but the Internet either outgrew them, or the fixes caused issues in other areas. Only one of these solutions is considered the ultimate fix to the original problem of running out of IPs. RFC 2460 protocol yields 340 undecillion, or 340 trillion trillion trillion, IP Addresses. The IPv6 specifications were written to mimic IPv4 while capitalizing on the years of experience and shortcomings of the IPv4. Below is a chart that compares the two protocols.

Table 3. Comparison of IPv4 and IPv6

	IPv4	IPv6
Length in bits	32	128
Maximum IPs	4,294,967,296	340,282,366,920,938,463,463,374,607,431,768,211,456
Format	4 octets	8 4 digit Hexadecimal integers
Notation	Binary	Hexadecimal
Subnetting	Yes	Not needed
Private IP	Yes RFC1918	LinkLocal
Loopback	127.0.0.1	::1

Benefits to IPv6 addresses are the obvious, a significant amount of more IPs, as well as the flexibility of being able to do more with the addresses.

2.5: Device Addressing

Devices that are communicating on an IP Network must have an IP address. An IP address can be obtained in two primary ways; one is by static configuration and the other is dynamically. Both methods have their benefits and drawbacks. Static requires manual intervention on each device, whereas dynamic allows for a device to automatically obtain an IP at the device's request, usually upon boot process. Dynamic requires an initial investment to prepare for this method, such as some configurations and software on the network to permit this setup. Using a dynamic method is most common and allows for easy and quick configuration to get a device communicating as quickly as possible. The most commonly used method of dynamic configuration is known as Dynamic Host Configuration Protocol (DHCP).

When comparing static method between the two versions of the IP protocol you will find that nothing prominent has changed; therefore, this research will focus on the similarities, differences, and additions between DHCPv4 and DHCPv6.

Table 4. DHCPv4 and DHCPv6 Comparison

Features	IPv4	IPv6	Benefits
Managed configuration flag	not avail	The router using router advertisements flags controls whether the nodes can use DHCP or not.	Configuration of nodes can be managed by a network policy.
Destination address of initial request	Broadcast	Multicast to all DHCP servers	More specific, more efficient, less overhead on link
Source address of initial request	0.0.0.0	Link-local address of the client	More specific, more efficient, less overhead on link
DHCP relay	need static list of DHCP	May use the all DHCP servers on the site multicast	More fault tolerant, and redundant, and

	servers	address	automated
Reconfiguration message	not avail	Server can request clients update their configuration(s)	Easier to do site/organization wide changes
Identity association	not avail	Clients can handle multiple DHCP servers and receive multiple addresses	More scalable use of DHCP
Dual Stack Transition Mechanism	not applicable	DHCPv6 used to give temporary IPv4 addresses in the DSTM transition mechanism	Efficient use of remaining IPv4 address space.

Table 4 compares IPv4 and IPv6 DHCP server features. For the most part, IPv6 DHCP remains that same with a few very nicely complimented additions and tweaks to increase efficiency. IPv6 DHCP (DHCPv6) has added the capability to signal DHCP clients to renew their IP [17]. Signaling a DHCPv6 client is known as a reconfig-init message. In IPv4 there was no way to tell a client to update its IP. This feature is extremely helpful when a network needs to be re-IPed. Also, when a client requests an IP, the client does not send the request to a broadcast IP, as in IPv4, but to a multicast address with known DHCP servers [9]. This change reduces the traffic on a network by limiting the destinations from everyone to a few, or in most cases, one.

IPv4 attempted to implement a hybrid method of static and dynamic but was flawed due to other protocol modifications interfering with its operation. This method was known as Automatic Private Internet Protocol Address (APIPA) [18]. This was the process of automatically obtaining an IP without a DHCP server on the network. The problem with APIPA was that it assigned an IP Address that was not part of RFC 1918's Private Address block. The IP that APIPA assigned began with 169. APIPA IPs had difficulty communicating with a LAN. IPv6 capitalized on this method by making slight modifications that repaired the functionality. IPv6's version of APIPA is referred to as Auto-Configuration [19]. Auto-Configuration uses a combination of special IPv6 addresses, along with a link layer address, such as a Media Access

Control (MAC) address. A MAC address is a unique identifier assigned to a network interface card. IPv4's Auto Configuration method expected conflicts and had mechanisms in place to deal with duplicates [18]. With IPv6 using a MAC address as a unique identifier assigned to a network interface card, it is able to avoid any duplicates on the network.

2.6: Naming

As IPv4 evolved, methods were implemented to make it easier to discover and navigate resources. One of those additions was the process of cross-referencing human-friendly names with an IP address. This process is known as Domain Naming Services (DNS) [20]. An example of this process is when a client attempts to go to a site like yahoo.com. The device will query a server that stores the reference table that contains the IP address. The server will respond with the IP Address. In the yahoo.com example, the DNS server would return an IP address such as 206.190.36.45. In IPv4 the name to IP address mapping is known as the 'A' record. IPv6 uses the term 'AAAA' record. Like the IP addresses of IPv6 the term 'AAAA' record is four times larger than IPv4 [18].

It is very likely that DNS contributed heavily to the success of the internet by making it easier to use. DNS is important in IPv4, but due to IPv6 addresses' more complex nomenclature, DNSv6 is more critical. Remembering a 32 bit decimal is much easier than remembering a 128 bit hexadecimal. According to Marshall and Crawford, if DNS were to break, the Internet would shut down [21].

2.7: Security

Security is arguably the most important topic of today's Internet. Everyone and everything touches the Internet. With the huge popularity and ubiquity of the Internet it was only a matter of time before the Internet became victim to malicious and illegal actions. To no

surprise IPv4 was not designed to combat security. The developers had implicit trusts between networks and resources [9]. It was not until many years later that IPv4 was retrofitted for security. In November of 1998 RFC 2401 IP Security Protocol (IPsec) was published and later was updated and replaced with RFC 4301. IPsec protocol included a suite of protocols that attempt to cover the majority of security scenarios. A few of the major protocols are described below.

Encapsulating Security Payload (ESP) – IPsec Protocol for encrypting data would take the datagram and wrap it in an encrypted datagram.

Authentication Headers (AH) – AH satisfies the permission control by providing integrity and authentication.

Internet Key Exchange (IKE) – IKE provides key exchange mechanism for two hosts to authenticate and transmit encrypted data.

Security Association (SA) – SA was the policy agreed upon between two peers via a key exchange or an authentication mechanism.

IPv4 would take a combination of the above along with other security measures to combat malicious attempts as well as secure confidential data. As with many other 'fixes' this was not a native technique and introduced many interoperability and performance issues. Once again, learning from the lessons experienced during the IPv4 protocols maturity, IPv6 built the protocol with security a mandatory and native component. The designers used the IPsec protocol suite for IPv6 and intertwined it into the IPv6 stack.

2.8: Conclusion

As seen in the previous sections, IPv6 has not only leveraged the successes of IPv4, as well as addressed the major issues with IPv4; it has also added great enhancements and

improvements. The foremost reason for the new protocol is not only the exponential increase in the amount of addresses, but key enhancements like native IPsec, fixed length IP header, and ease of re-IPing will prove to be very valuable in the future of IPv6.

Although IPv6 was built with IPv4 in mind, IPv6 is not an upgrade to IPv4. IPv6 is a separate protocol. IPv6 is not backwards compatible. Therefore, transitioning to the new protocol will require planning. The following section will investigate the mechanisms available to transition to this new protocol.

Chapter 3: Methodology

3.1: Introduction

In order for the Internet to continue to grow and develop the address space must be increased. As of April 19, 2011, Asia Pacific Network Information Center (APNIC) handed out their final block of Class A addresses. American Registry for Internet Numbers (ARIN) projects they will have exhausted their Class A addresses by March 2015. [22] New IPv4 Addresses cannot be created. The protocol must be replaced. IPv6 has been the accepted successor to IPv4 and provides a much larger amount of IP Addresses. The IPv6 standard was published in the '90s, and is yet to be largely deployed. One of the primary reasons IPv6 has not been fully installed is due to the overwhelming presence of IPv4 across the globe. Among the approximate 3.5 billion devices on the Internet a large amount are mission critical systems, such as banking, email, healthcare, and governmental systems [23]. The uptime on these systems is critical and users will not tolerate any interruption in service(s) [24]. Society has grown to know the Internet as a reliable, growing, evolving entity. Careful planning must be considered when developing a deployment plan. Due to the sheer scale of the project it is apparent that the transition cannot be done instantaneously. In order to cause as little interference as possible the transition will have to be slowly introduced into production. Many strategies and techniques will be leveraged to make this happen. The complete transition will take many years, if not decades to complete. Some experts say that the transition will carry on well into the 22nd century [25]; and during this time we will remain heavily reliant on IPv4.

Each network accessing the Internet is unique and must design the migration path specific to its configuration. Legacy systems that will not have IPv6 support need to have solutions designed to continue the uninterrupted access. The following sections will discuss a

few of the more popular industry standard transition techniques, their advantages and disadvantages, the primary application of each, and how they may be used together in order to provide a more complete solution.

3.2: Techniques

There are three main approaches to transitioning. Dual stack (DS) is a parallel approach that is more than likely the primary transition method as an accepted starting point. As the name indicates a device, computer, printer, mobile, switch, or router has support for both protocols, IPv4 and IPv6. DS allows a device to be able to send and receive either IPv4 or IPv6 packets. Dual stack IPv4 and IPv6 method can be compared to the methods used when TCP/IPv4 and IPX were competing for ubiquity in the mid 90's. A major benefit of Dual Stack is that a device will receive all traffic intended for it regardless of whether or not the source sent it using the correct protocol. A disadvantage of DS is that the device running two protocols may require double the processing cycles. According to a Network World survey, the primary reason for companies to migrate is address concerns and they suggest the use of a dual stack approach [26].

Translation is another popular method. Translation techniques are the process of inserting a translation device between the IPv4 and IPv6 networks. The device will convert IPv6 packet to IPv4 packets, and vice versa. This technique is used when an IPv6 host wants to communicate with many IPv4 hosts [27]. This technique can be compared to the NAT protocol for IPv4.

Drawbacks to this technique are:

- **Single points of failure** – when network address translations occur the traffic's return path must be the same as its initial path, because the device that translated the packets is the only device that knows what to translate it back to and where to relay it to.

- **Potential bottleneck of traffic** – More processing is needed to do translations.
- **Application impact** – If the data needs to be encrypted from end to end, this technique cannot achieve this due to the translation must modify the packets during the translation, breaking the integrity of the encrypted packet [9].

Translation is the least desirable method do to its downfalls and should only be used in a last resort scenario. Translation should be short term until another solution is implemented [9].

The third of the three most popular transition techniques is Tunneling. Tunneling is the process of inserting the packet into the native protocol packet, e.g. inserting IPv6 packet into an IPv4 packet [9]. Benefits of tunneling are organizations can easily connect IPv6 networks over IPv4 infrastructures. This is beneficial when an entity has no control of the network(s) in between the IPv6 networks. Tunneling also has many downsides, for example an IPv4 networks cannot access IPv6 networks natively and must add other techniques to interact with each other. Another major disadvantage is that tunneling makes troubleshooting much more difficult because of the extra layers of complexity [9].

Of the three approaches discussed, the dual stack technique is preferred, easy to use and flexible. When everything is upgraded to IPv6, IPv4 can simply be disabled. Additionally, dual stack is the foundation of other mechanisms like tunneling. Disadvantages to dual stack are (1) the device is running two protocols and requires more CPU, memory, multiple internal tables, (2) routing and routers need to run multiple protocols, again requiring more processing power. However, for most conversions dual stack is the preferred method. Tunneling requires little intervention from 3rd parties, such as an ISP, and backbones do not need to be upgraded immediately. Disadvantage to tunneling are CPU load on router, tunnel entry/exit points need more than normal time to process, CPU to encapsulate/decapsulate, single points of failure,

troubleshooting is more difficult due to hop count/MTU size/fragmentation issues. Translation should be used only if other techniques are not possible, and should be temporary as one move toward the capability to use other techniques. A major disadvantage to translation is losing the capability to use IPv6 enhancements, such as end to end security, topology restrictions due to replies must come through the originating NAT device producing a single point of failure. An advantage to translation is that it allows IPv6 devices to communicate directly with IPv4 devices and vice versa; but this advantage is not a long term solution.

3.3: Conclusion

Given all the options available, there is not one solution for all scenarios, but many solutions for one scenario can arise. As with all technology the solution must be tailored to the environment. Network architects/engineers must apply many of the above techniques to provide and migrate to IPv6. For the current study, dual stack will be used with a strategy known as the "edge in" strategy will be used. The "edge in" strategy is the process of starting at the edge of the network and working toward the core [28]. Starting from the edge will provide minimum interruption, low impact solution to an organization. Allowing the business to continue to run and function while the IT department slowly migrate system(s) to a Dual Stack configuration, which is essential to a business's survival.

3.4: Simulated Environment Architecture

The following section will document and analyze the process(es) of taking a simulated Small to Medium sized Business Local Area Network and migrate it to an IPv6 network. The network will include the following:

- Windows 2008 Active Directory and DNS
- Windows 2008 DHCP server that will also have File and Print services

- Windows 2008 Web server
- Window 7 laptop
- Window 8.1 laptop
- Cisco 3560 Layer 3 switch
- HP Color LaserJet printer
- Linksys Business Series Model WAP200 Access Point

All of the above equipment will be built from scratch taking all default settings. Each Windows OS will have Wireshark installed to allow packet capturing. Wireshark is an application that captures packets and presents them in a GUI for low level analysis. Wireshark will enable the ability to see specific frame and packet information, such as the protocol being used or the data in the payload.

Initially, this network will be installed and configured as an IPv4 LAN. The Cisco 3560 will have multiple Virtual LANs (VLAN) configured. Devices will be placed on different VLANs to better simulate larger, more complex network such as in the real world. All services, such as Active Directory (AD) will be utilized and tested. A simple webpage will be placed on the web server, files will be shared among systems and print jobs will be sent to the printer. When the network build has been completed and tested the conversion to IPv6 will start. The conversion will leverage the "edge-in" strategy with the laptops being the starting point. As more systems are configured for Dual Stack, services will be added and/or upgraded to assist in the simplification of administration, such as DHCPv6 will be added to the environment to assist with IP addressing. It is not uncommon that a mid-sized business has several systems that need IP addresses given to it automatically. DHCP will help ease administration of this process. DNSv6 will be configured to assist in the IP to name conversion. IPv6 IP addresses notation is

much more complex than IPv4 addresses, resulting in DNS being much more critical to continuity.

During this conversion, many tools will be used to confirm the desired results are achieved. Some of the tools that will be used are:

- ipconfig – ipconfig tool will be used to view and configure a systems IP settings.
- Wireshark – Wireshark will be used to confirm the proper results are being communicated between systems.
- NSlookup – nslookup will be used to test DNS resolution is functioning as expected.
- Sh arp, sh run, sh int – Cisco commands that will confirm configurations are IPv4 or IPv6 as expected.

The goal of the experiment is to provide a case study for the conversion of IPv4 to IPv6 in which the entire network will have a deployed and documented IPv6 network.

3.5: Anticipated Challenges

During the conversion of the network, there are a number of anticipated challenges that may arise. The following challenges are anticipated during deployment.

3.5.1: IP Scheme

Deciding on an IP scheme will present itself as a challenge. Having little interaction with the IPv6 protocol will give rise to unfamiliar territory. When deciding on the IP Scheme, questions like the following will arise:

- How will the LAN be segmented

- Where will the IP address Network Prefix stop and where will the Host portion of the address start.

3.5.2: Device Addressing

Another challenge that was predict is with device addressing. Will it be wise to use Static (manual) addressing, or auto? Auto addressing is the natural choice, but when using auto addressing more options become available; like whether to use the native auto-configuration of IPv6 or do we introduce DHCP?

3.5.3: DNS

The third anticipated challenge has to deal with DNS. The Internet has become dependent on DNS, and IPv6's address complexity magnifies the need for DNS functionality. Will the DNS services need to be upgraded, or modified to allow the AAAA records?

Chapter 4: Experiment

4.1: Introduction

To underst and the conversion of IPv4 to IPv6 in a small to medium size business (SMB), an experiment was conducted which included 3 phases. At the end of each phase, a test plan was performed. First, a simulated network consisting of items typically found in a SMB (e.g. Active Directory, VLANs, wireless access, file & printing services and network infrastructure services) was built with all network equipment running the IPv4 protocol. Once the build phase was completed and defined use cases were tested, the next step conducted was to configure dual stack on each network node. To further simulate the process of an organization migrating to a new protocol, the configuration of dual stack began in an area or section of the network with a low impact to production. This strategy is referred to as an 'edge-in' approach. Using an edge-in approach can reduce critical business process interruptions by making changes to a system with low to no impact on a company's business continuity. When all the nodes were configured for a dual stack protocol, thorough testing to confirm all services are still available was performed. After successful testing, the last step was the removal of IPv4 protocol from the dual stacked equipment. When the network was solely IPv6, final testing phase was conducted. In the subsequent sections, the details of each phase are discussed.

4.2: Phase 1 - Network Build

Phase one consisted of building the lab in which the IPv6 conversion will be simulated. The lab consisted of:

- Windows 2008 R2 Active Directory and DNS
- Windows 2008 R2 DHCP server that will also have File and Print services

- Windows 2008 R2 Web server
- Window 7 laptop
- Window 8.1 laptop
- Cisco 3560 Layer 3 switch
- HP Color LaserJet printer
- Linksys Business Series Model WAP200 Access Point

Considering Windows XP is end of support as of April 8, 2014 [29] and 2003 is End of Support on July 7, 2015 [30], these operating systems were not included in the network. Windows 7 & 8 as well as Windows 2008 were chosen as they are more likely be found in a corporate environment. Each Windows operating system was installed from scratch taking all default settings. According to Microsoft [31], Windows 2008, 7 and 8 are IPv6 ready with the service enabled by default. There are many debates about whether or not to disable the service when an organization is not using the IPv6. The assumption is that having both will increase unneeded network traffic, complicate DNS, tax processing on clients running both protocols. For this research, the industry standard practice will be followed, which is to disable IPv6 services until needed.

4.2.1: Switch Configuration

The first device configured was the Cisco WS-C3560-8PC running C3560-IPBASE-M version 12.2(35)SE5. The switch was configured via a console connection with the minimal changes that provided layer 3 capabilities. For a complete inventory of the switch, refer to Appendix A for the '*sh ver'* command output; as well as a full configuration. The L3 switch's hostname is "ipv4ipv6". Telnet was configured to allow access from all systems with minimal intervention. The telnet and enable mode passwords were both set to 'cisco'. To simulate a

larger scale network and achieve simple Layer 3 communications 6 VLANs and VLAN interfaces were configured as follows:

Table 5. VLANs on Switch

VLAN #	Network	VLAN Interface IP (Gateway)	Purpose
1	10.1.1.0/24	10.1.1.1	Windows AD/DNS services
2	10.2.1.0/24	10.2.1.1	DHCP, file and print
3	10.3.1.0/24	10.3.1.1	Windows Web server
4	10.4.1.0/24	10.4.1.1	Wireless Client Network
5	10.5.1.0/24	10.5.1.1	Wired Client Network
6	10.6.1.0/24	10.6.1.1	Printer VLAN

Since this switch will need to forward (route) packets onto other networks, the *ip routing* command was injected. This will enable packet forwarding. Each physical port on the switch will be assigned to a different VLAN. Port 1 will be part of VLAN 1; Port 2 will be part of VLAN 2, and so on. See Table 6.

Table 6. Port/VLAN Assignment

Port	VLAN	Purpose
1	1	AD/DNS
2	2	DHCP/File & Print Services
3	3	Web Server
4	4	Wireless Access Point
5	5	Wired Network
6	6	Printer Network
7	5	Wired Network

4.2.2: Server Configuration

The first 2008 server was called 2008-1. Once the OS was successfully installed the IPv6 protocol was uninstalled and the following IPv4 settings found in Table 7 were configured. The server was then connected to the network switch.

Table 7. 2008-1 IP Settings

IP Address:	10.1.1.2
Subnet Mask:	255.255.255.0

Gateway IP Address:	10.1.1.1
DNS Server(s) IP(s):	None (no DNS server at this point)

Simple network connectivity was confirmed by "pinging" the gateway IP address using the following command via the command line tool:

C:\ping 10.1.1.1

Active Directory Services (ADS) were installed via the Server Role Configuration Tool (see Appendix B for a screenshot of the Server Configuration Tool). Once the services were installed, the *dcpromo* command was run. The *dcpromo* command initiates the Wizard to promote the server to a Domain Controller. Installation of Active Directory requires DNS services; therefore, during the installation and configuration of ADS, the DNS services were installed and configured. The name given to the domain was "ipv6.lab"

At this point the DNS server settings for the 2008-1 DNS Client were auto configured to 127.0.0.1, the local loopback IP address. The loopback IP for the DNS Client settings were retained.

Now that the system has a complete AD environment, accounts can be created and computers can be added. The following user accounts were created and assigned roles.

Table 8. Domain Accounts

Account	Roles
Administrator	Domain/Enterprise Admin
Mabate	Domain User

The next node configured was 2008-2. Once again, Windows 2008 R2 was installed using defaults. IPv6 was uninstalled and the IPv4 settings found in Table 9 were used. The server was connected to the network switch.

Table 9. 2008-2 IP Settings

IP Address:	10.2.1.2
Subnet Mask:	255.255.255.0
Gateway IP Address:	10.2.1.1
DNS Server(s) IP(s):	10.1.1.2

Simple network connectivity was tested, and the server was added to the ipv6.lab domain. See Appendix B for a screenshot of the systems on the domain. Using the server configuration manager, the DHCP, File Service and Print service roles were added to this server.

The following DHCPv4 Scopes were created:

Table 10. IPv4 DHCP Scopes

VLAN	IP Range	DNS Server IP	Gateway IP
4	10.4.1.10-10.4.1.50	10.1.1.2	10.4.1.1
5	10.5.1.10-10.5.1.50	10.1.1.2	10.5.1.1

During the DHCP setup the server asked if this DHCP server would need DHCPv6 in Stateless Mode. At this time, IPv6 services were not required, so the response was "no". See Appendix C.

4.2.3: Additional Microsoft Infrastructure Services Configuration

File services were configured and a Distributed File System (DFS) was added. The DFS share was called "ipv6labdfs". The permissions were modified on the share to allow the built-in "admins" and 'users' groups to be able to Read and Write. See Appendix C. The DFS share will be utilized throughout this experiment by placing all logs from each system in the share; allowing for constant testing throughout this lab. See Appendix C.

Print services were installed successfully, and the HP Color Laser Printer model CP1518ni was configured with the settings found in table …. The printer was then added to the Print server via the "Network Printer Installation Wizard". A TCP/IP device was selected and

the IP address and port name was: 10.6.1.2 (This may cause issues later considering IPv4 is hard coded)

Table 11. Printer IPv4 Settings

IP Address:	10.6.1.2
Subnet Mask:	255.255.255.0
Gateway IP Address:	10.6.1.1
DNS Server(s) IP(s):	Not needed

The printer was shared as "Printer" and published to the Active Directory. The printer was added to 2008-1 via the newly created share and a test print occurred.

2008-3, another Windows 2008 R2 system was created. The server was connected to the network switch and the following IP settings were used.

Table 12. 2008-3 IP Settings

IP Address:	10.3.1.2
Subnet Mask:	255.255.255.0
Gateway IP Address:	10.3.1.1
DNS Server(s) IP(s):	10.1.1.2

Once network connectivity was confirmed the system was added to the ipv6.lab domain. After restart the IIS server role was installed via the Server Configuration Tool. After the IIS service installation completed the webpage http://2008-3 was available. See Appendix D.

4.2.4: Access Point Configuration

A Linksys WAP200 Business Series Access Point was connected to the network via switch port 4 on VLAN4. The default configuration sets the IP to 192.168.1.245/24. To configure the AP for this network we temporarily changed and IP of a server to reside on the 192.168.1.1/24 network. Once they were in the same broadcast domain, communications were able to flow to the WAP without a router or routes needing configured. The AP was configured via the web browser interface, using the following settings.

Table 13. Access Point Settings

IP Address:	10.4.1.2
Subnet Mask:	255.255.255.0
Gateway IP Address:	10.4.1.1
DNS Server(s) IP(s):	Not needed

The AP rebooted. The server was reconfigured for the proper network. Once the AP rebooted, access was regained via the new IP address. At this time the Wireless was configured. The SSID that was broadcasted was IPv4IPv6Lab. Security was WPA2Personal and the passphrase was ipv6lab1234.

4.2.5: Client Configuration

The client OS systems were built. All client installs took default settings. One of the Win7 systems, called Win7-2, was connected to the ipv6lab SSID. The other two systems, Win7-1 and Win81 were hard wired to the appropriate switch port(s) which are members of VLAN5.

As suspected, upon initial boot of the clients an attempt to obtain an IP via a DHCP server failed. The reason for this is because the client systems do not have a DHCP server within their broadcast domains. To resolve this issue, DHCP relay agents were configured on the VLANs. This was achieved by configuring the Cisco Switch as follows:

Conf t

int vlan 4

ip helper-address 10.2.1.2

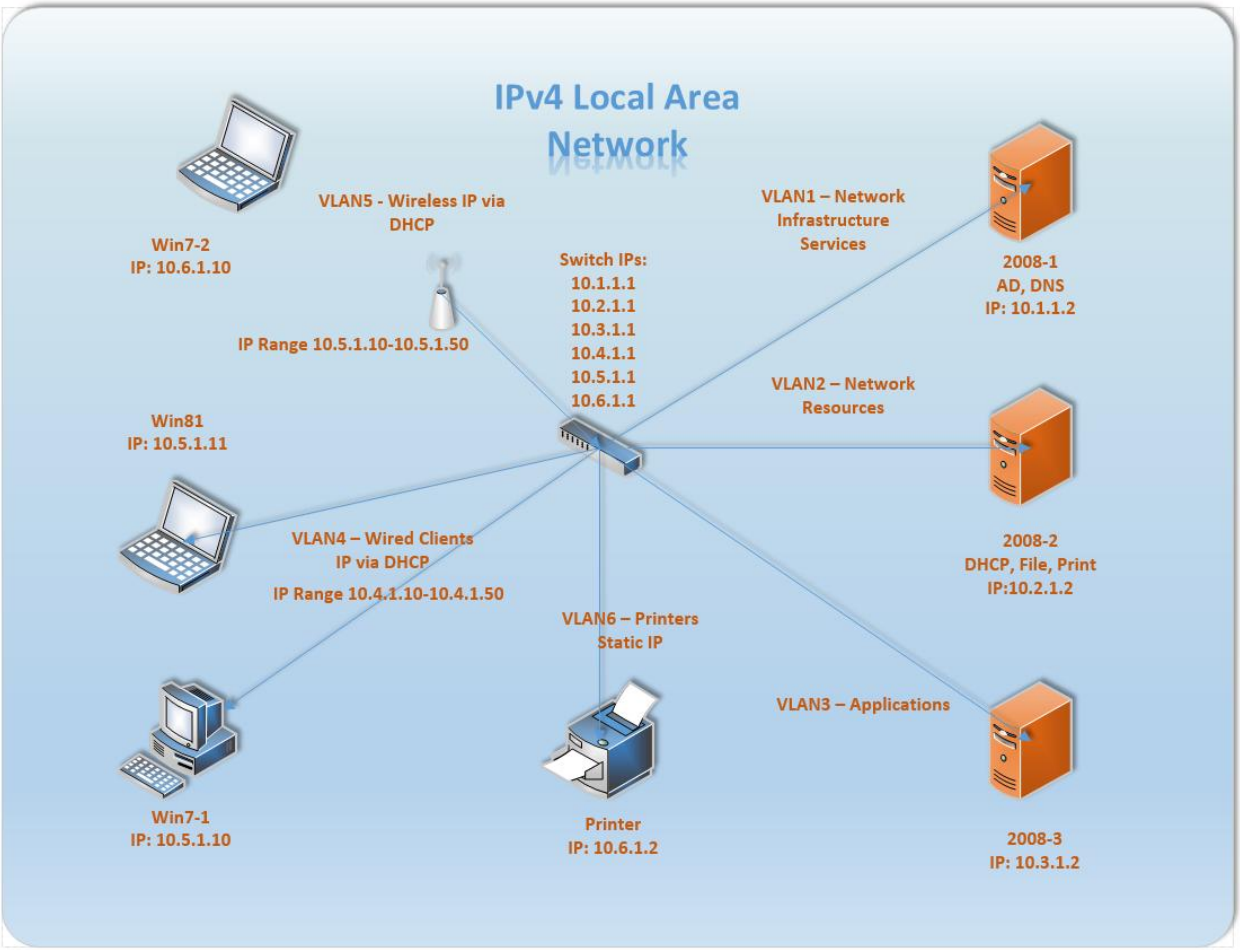
Int vlan 5

ip helper-address 10.2.1.2

The *ip helper-address* tells the switch to forward any DHCP request to 10.2.1.2, the DHCP server. Once the *ipconfig /renew* command was executed on the clients IP address information was automatically obtained. The three clients were added to the domain and as you can see from Appendix B the clients show up on the domain.

At this point the network looks like figure. The full switch configuration can be found in Appendix A:

Diagram 1. IPv4 LAN – IPv4 Only Topology



Following the test plan in Appendix E, tests were run across the entire network. All devices were running the IPv4 protocol without any communication issues between the devices. Appendix E has the complete results of the initial network tests.

4.3: Phase 2 - Dual Stack Configuration using "Edge-In" Approach

Once the network was successfully up and running on the IPv4 protocol, dual stack configuration began. As previously mentioned, this project used an 'edge-in' approach with the goal being to slowly step into the new protocol and learn. In simulating a live environment, the first objective was to get two minimal impact systems communicating with each other over the IPv6 protocol. Then, applying what was learned to the rest of the network nodes during the migration process.

4.3.1: Clients on same link Dual Stack Configurations

Starting with a client computer (Win7-1), IPv6 protocol was enabled on the network adapter connected to the IPv4IPv6Lab LAN. As the IPv6 protocol was enabled, the NIC immediately performed the auto configuration process for the link local address. Link local, comparable to APIPA [18], is a solution for auto IP addressing a network without a DHCP server present. The link local is distinguishable by the prefix of fe80. To avoid IP conflicts, auto-configuration uses the NIC's layer 2 address, also known as the media access control (MAC), and embeds it in the host portion of the IPv6 address (e.g.fe80:0000:0000:6553:67a1:6b69:7291). This process is called Interface Identifier [32]. Now that there was a node on the network with an IPv6 address lots of IPv6 traffic was flowing on the Wireshark capture(s).

The next node to be dual stacked was the laptop client (Win81) with Windows 8.1 operating system. The reason this system was selected was because it is on the same VLAN as the WIN7-1, which we just dual stacked. This will continue the strategy of edge-in by approaching the easier elements of the network first and then work our way into the more

difficult areas, such as systems on different VLANs. Systems on different VLANs will require IPv6 packet forwarding.

Enabling IPv6 protocol stack on Win81 is practically the same as Win7-1. Like the previously stacked client, once IPv6 was enabled, the Link-Local IPv6 address was created. Now that there are two systems on the network with IPv6 addresses, testing was conducted to communicate with each other. Using the link local addresses, the two systems were able to communicate with each other. The current network topology looks like Diagram 2.

Diagram 2. IPv6 LAN – 2 Nodes Configured with IPv6

4.3.2: A Client on a different link Dual Stack configuration

The next system to be dual stacked resided on a different network from the current dual stacked systems. When it was dual stacked it was be unable to communicate with the other IPv6 clients via a link local IP address. This is because, as discussed in Chapter 2 addressing, Link local addresses are restricted to the link layer. In order for the IPv6 communications to traverse the VLANs two things needed to be accomplished:

1. An IP addresses scheme needed to be developed.
2. The L3 switch needed to be configured with IPv6 addresses, and packet forwarding needed to be enabled and configured.

Deciding on an IP address scheme depends on what type of addresses used. As discussed in Chapter 2 there were a few options.

- Link local - which will not work for this environment because systems need to communicate with other links.
- Global Unicast Addresses – these are comparable to IPv4 public IPs.
- Unique Local IPv6 Unicast addresses (formerly known as Site addresses) these are comparable to the IP Private address space, e.g. 10.0.0.0

The network can continue the IPv4 methodology and use Unique Local Address (ULA) as we do for private networks today, but then later projects will have to use NAT at the Internet edge to communicate with other public networks. One argument is presented by Ed Horley, a leading expert on IPv6 administration, who suggests we need to start thinking in terms of IPv6 where there is no longer an IP shortage [33]. Administrators should embrace IPv6's large address space and remove the extra level of complexity known as NAT. Removing NAT will restore true end to end communications; eliminate special handling, and potential application

issues. He also shoots down the theory that NAT gives a level of security because of "hidden" IPs.

After researching the IP addressing options, the choice was made to use Global Unicast addresses for this project and utilize the grand IPv6 address space. Typically one would obtain these addresses by contacting an ISP. The ISP would get them from a Regional Internet Registry (RIR) and the RIR would get them from the IANA. For the simulated network, these steps were skipped because this network will not need to communicate with the Internet.

For this network, the following randomly selected global unicast address prefix was used. The first 48 bits are the prefix provided by the ISP.

2001:0db8:85a3

The next 16 bits are for Site subnets.

:0001

The last 64 bits are for the host(s).

Table 14 depicts this network's IPv6 scheme and cross references it to the current IPv4 network.

Table 14. IPv4/IPv6 Network Address Cross Reference Table

IPv6 Prefix	IPv6 Subnet	IPv4 Network
2001:0db8:85a3	0001	10.1.0.0
2001:0db8:85a3	0002	10.2.0.0
2001:0db8:85a3	0003	10.3.0.0
2001:0db8:85a3	0004	10.4.0.0
2001:0db8:85a3	0005	10.5.0.0
2001:0db8:85a3	0006	10.6.0.0

Once the IP Scheme was decided, the L3 switch was configured. The following table takes the L3 switch IPv4 IPs and cross references them with the IPv6 to be assigned.

Table 15. IPv4/IPv6 VLAN Table

VLAN	IPv4 Address	IPv6 Address
1	10.1.1.1/24	2001:0db8:85a3:0001:0000:0000:0001/64
2	10.2.1.1/24	2001:0db8:85a3:0002:0000:0000:0001/64
3	10.3.1.1/24	2001:0db8:85a3:0003:0000:0000:0001/64
4	10.4.1.1/24	2001:0db8:85a3:0004:0000:0000:0001/64
5	10.5.1.1/24	2001:0db8:85a3:0005:0000:0000:0001/64
6	10.6.1.1/24	2001:0db8:85a3:0006:0000:0000:0001/64

The following commands were run on the switch:

(config-if)#ipv6 address ?

% Unrecognized command

(config)#ipv6 address

% Invalid input detected at '^' marker.

However, when the commands were attempted, an error occurred.

According to the article, "Where is IPv6 on my Cisco 3560 in my CCNA lab?" on www.certificationkits.com [34], the switch's loaded Switch Database Management (SDM) template needed to be changed. When following the source's directions, it was discovered that the ipv6 lab switch was in fact running the wrong template.

To resolve this issue, the following commands were executed:

(config)#sdm prefer dual-ipv4-and-ipv6 routing

End

Reload (confirm)

When the switch was available, the following commands were run on the L3 switch to configure the each VLAN interfaces with an IP address: (obviously substituting the X for the appropriate VLAN.

Conf t

int vlan <X>

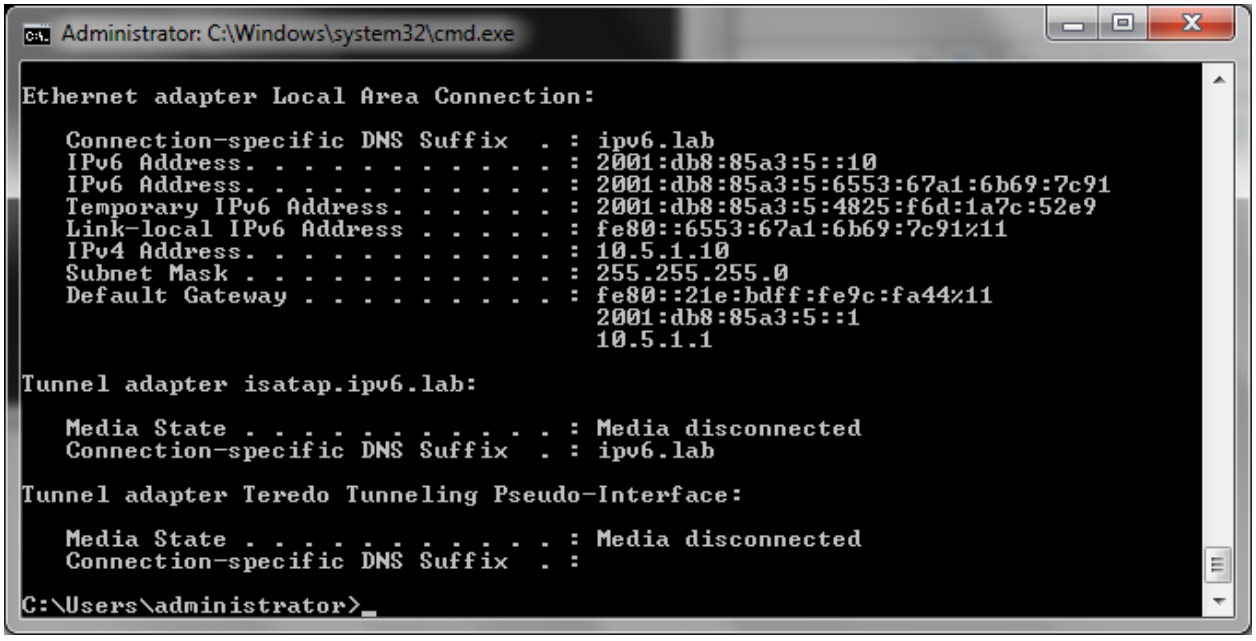
ipv6 address 2001:0db8:85a3:000X:0000:0000:0001/64

end

wr

As each VLAN was configured, the L3 switch started sending Router Advertisement ICMP messages to the network, multicasting the network prefix. This message enables the systems on the link to learn what network prefix to configure when auto-configuring their global unicast IPv6 address. The systems that are configured for IPv6 acknowledge the advertisement and began the auto-configuration process; still using the Unique Identifier process to configure the host portion of the address. Below is a screenshot (Figure 4) of the Win7-1's global unicast auto address.

Figure 4. Win7-1 IPconfig screenshot



Before the switch was configured with IPv6 global addresses, the *ipconfig* command only showed an auto configured Link Local IPv6 address. After the new configuration, the *ipconfig*

command showed an IPv6 address starting with a global unicast prefix, 2001:0db8:85a3. Like IPv4, IPv6 has the ability to have many IPv6 addresses. The auto-global IP is quite long and confusing. Another IPv6 address can be configured on the NIC that is relatively less complicated.

For example: 2001:0db8:85a3:0001:0000:0000:0010

And with some of the built in shortcuts of IPv6, the address can be more simplified. Using the IPv6 shortcut notation the above example would look like this: 2001:db8:85a3:0001::10 [9].

To take advantage of the shortcut notation, a manual IP address was configured. This involved manually configuring IPv6 addressing within this project as each system is dual stacked. Win7-1, Win7-2 and Win81 were manually configured with IPv6 addresses. (See Table 16 for IP assignments). IPv6 configuration is similar to IPv4 with the exception of the subnet prefix. The subnet prefix tells the network node where the host portion of the IP address begins. See Appendix F for a screenshot of the IPv6 manual configuration screen.

Table 16. Clients Manual IPv6 Addresses Table

Hostname	Manual IPv6 Address
Win7-1	2001:db8:85a3:0005::10 expanded 2001:0db8:85a3:0005:0000:0000:0000:0010
Win81	2001:db8:85a3:0005::11 expanded 2001:0db8:85a3:0005:0000:0000:0000:0011
Win7-2	2001:db8:85a3:0006::10 expanded 2001:0db8:85a3:0006:0000:0000:0000:0010

At this point, there are three systems configured with routable IPv6 addresses (i.e. Win7-1, Win7-2, and Win81). Tests were conducted to confirm connectivity was still available. Ping tests were performed from all three stations. The two systems on the same VLAN were still able to communicate, but the third system was still unable to communicate across the VLANs. This

result brought attention to the switch. The isolated system with an IPv6 protocol was able to ping its gateway, the L3 switch IP (2001:0db8:85a3:0006:0000:0000:0001) but not beyond that. With prior knowledge of IPv4 the solution to this issue is to enable IPv6 routing on the switch. This will enable the switch to forward (route) packets to other networks. To enable IPv6 routing, the following command must be entered into configuration mode on the switch:

Ipv6 unicast-routing

This command is very comparable to the IPv4 command of '*ip routing'*.

When injecting this command on the L3 switch the following error was received.

Command unrecognized

Upon researching this issue, it turned out that the IOS of the switch does not support this command set, and needed to be upgraded to IPServices [35]. Using a Cisco account, the proper IOS (c3560-ipservicesk9-mz.150-2.se5.bin) was downloaded from the Cisco support site.

Using a TFTP server application, the new IOS file was copied to the switch using the following commands:

copy tftp: flash:

Address or name of remote host []? 10.5.1.10

Source filename []? c3560-ipservicesk9-mz.150-2.SE5.bin

Destination filename [c3560-ipservicesk9-mz.150-2.SE5.bin]ac?

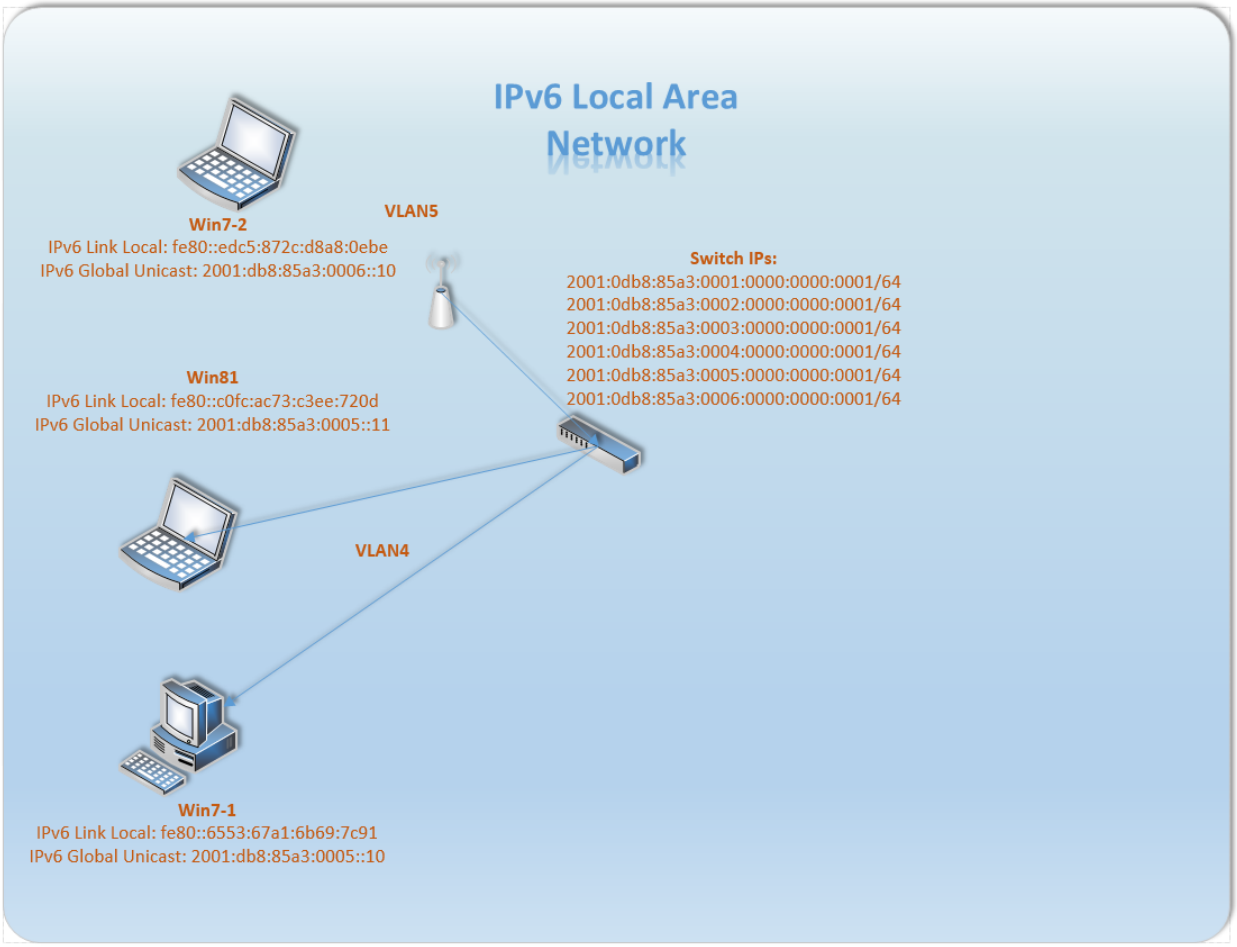
boot system c3560-ipservicesk9-mz.150-2.SE5.bin

wr

A reload command was initiated and the switch rebooted. When the switch completed the reboot the '*ipv6 unicast-routing'* command was available and executed. Communications

test across VLANs was successful. The following Diagram 3 shows the IPv6 network up to this point.

Diagram 3. IPv6 LAN – Nodes on Different VLANs Communicating Using IPv6



4.3.3: Dual Stack on Printer and Access Point

Continuing the edge in process, IPv6 services were enabled on the Printer. The HP printer has IPv6 support, but interestingly enough the printer only allows auto-configuration for an IPv6 address, this includes DHCP. It is suspected that if a reservation were on a DHCP server one could force the Printer to grab a specific IP address, and perhaps if the firmware were updated the Printer would allow manually configured IPs.

Next, the attempt to add the Linksys WAP200 Business series Access Point (AP) to the IPv6 network is performed. When attempting to configure IPv6 support on the AP via the management interface it was discovered that there is no support for management via IPv6. Researching and upgrading the firmware resulted in the conclusion that this device is not capable of being managed via an IPv6 IP address.

4.3.4: Dual Stack Servers

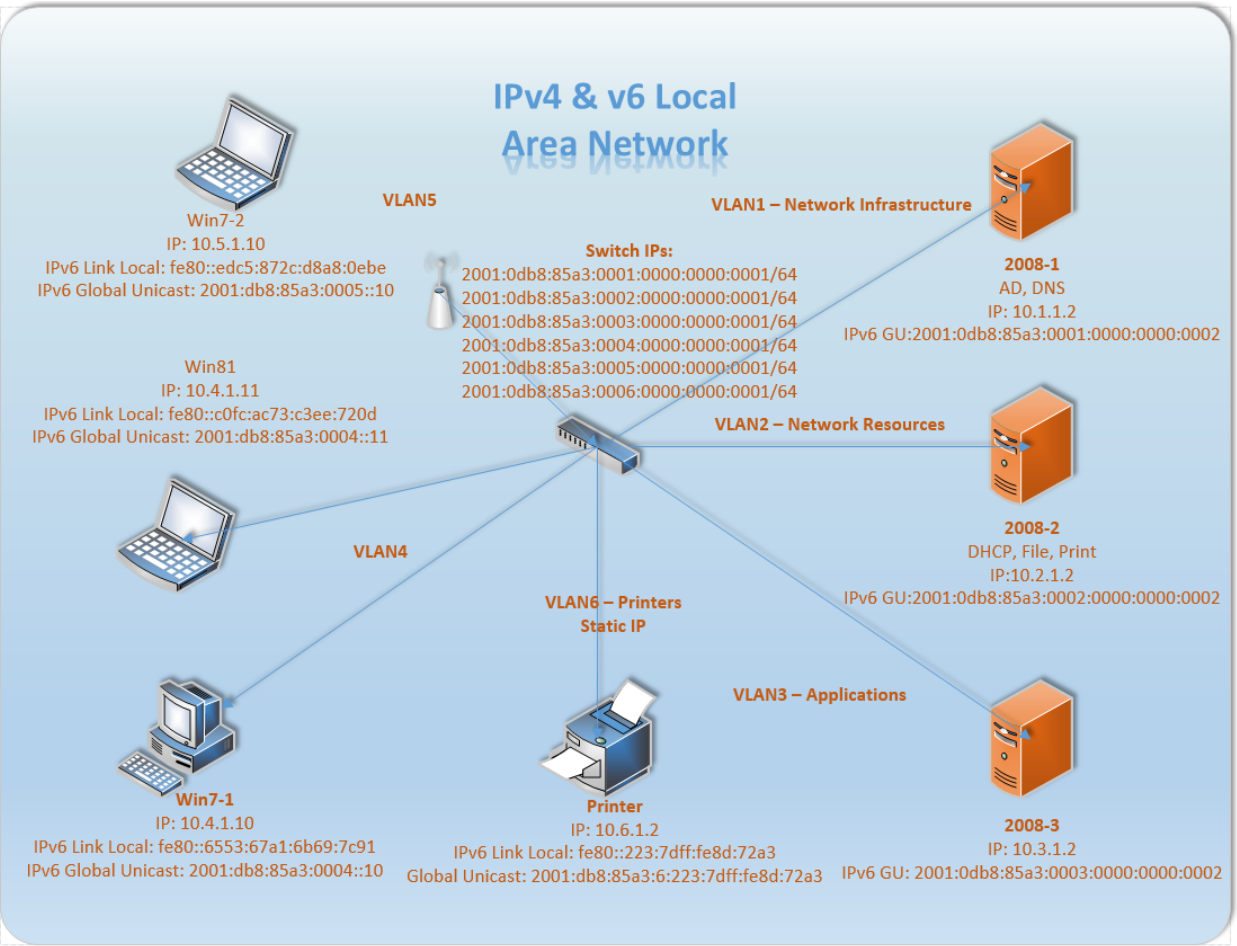
The nature of servers is to have a static IP address; therefore manually configuring the IPv6 IPs was completed. As with the other Windows systems, the servers also auto-configured a link-local and a global unicast IP address. Initial testing of connectivity via the link-local and global unicast addresses was successful. Considering these are servers and the IPv4 addresses are static, static IPv6 addresses were set on the servers. See Table 17 for static IPv6 addresses assigned to the servers.

Table 17. Server Manual IPv6 Addresses Table

Server Hostname	IPv4 Address	IPv6 Address
2008-1	10.1.1.2	2008:0db8:85a3:0001::2
2008-2	10.2.1.2	2008:0db8:85a3:0002::2
2008-3	10.3.1.2	2008:0db8:85a3:0003::2

At this point all systems have been configured with dual stacks. The network looks like Diagram 4.

Diagram 4. IPv4/IPv6 LAN – Full Dual Stack Topology



4.3.5: Configure Services to support both Protocols

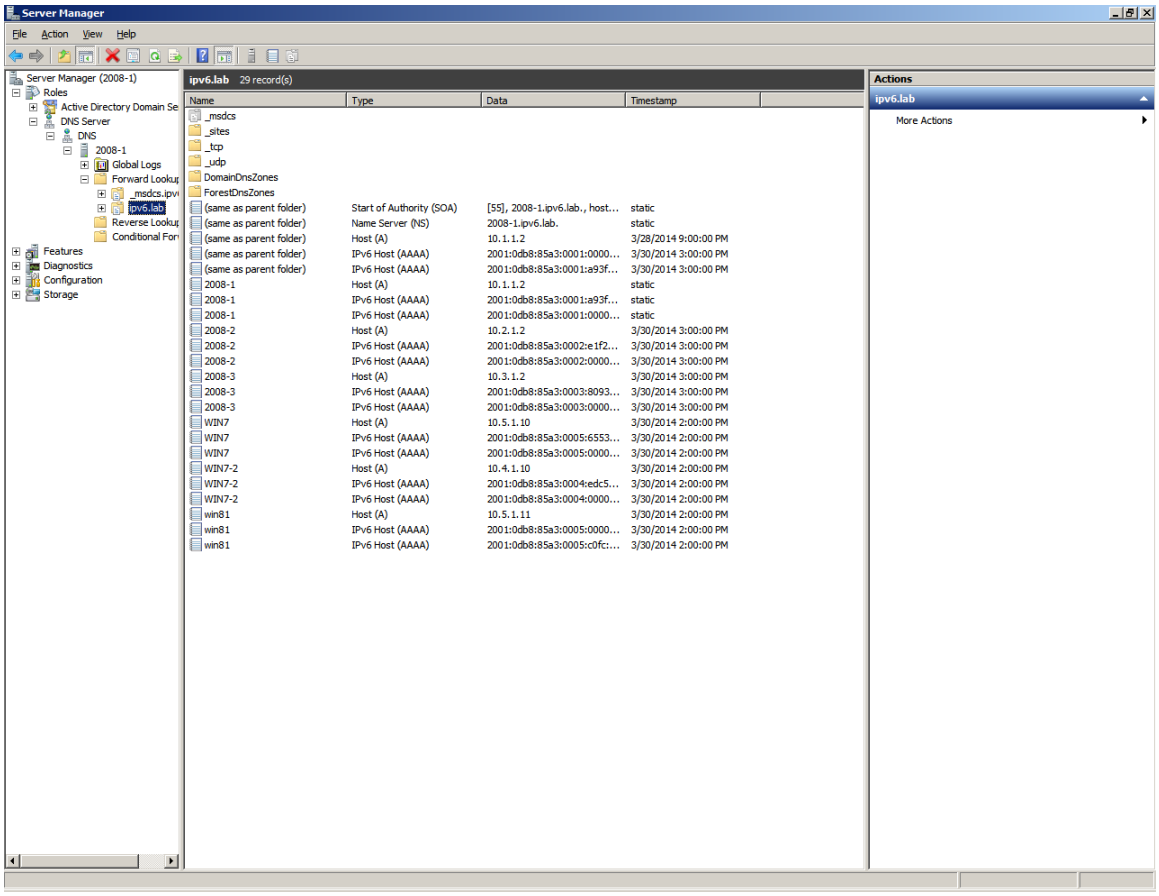
Now that all systems are capable of IPv6 it was now time to start testing and upgrading services. The first service to upgrade/test is the DNSv6.

4.3.5.1: DNS

It was initially thought that the DNSv6 service(s) would need to be installed. Apparently as the systems received a Global Unique address, whether it was auto or manual, the DNS clients on the systems registered the IPv6 address on the DNS server. This was made possible the instant the DNS server, 2008-1, was configured with an IPv6 address. This was confirmed by the following Network World article [36]. As seen from the Figure 5, the IPv6 addresses are

represented by an ‘AAAA’ record. DNS resolution tests were run to confirm the ability to resolve IPs.

Figure 5. DNSv6



4.3.5.2: DHCP

DHCP needs to be modified by adding the v6 functionality. Since 2008 already supports DHCPv6, all that was done at this point was add the scopes for the two DHCP Address ranges. When the options were being built, the DNS option was included in the scope. The DNS Recursion Server List option, option 23 is comparable to IPv4’s DNS Servers option. At this point the IPv6 address of the 2008-1 server was entered. When this IP was added a validation to see if that address was a DNS server was performed. The action took a little while and

responded with an error but allowed the process to proceed with the IP. The error received that stated the IP address provided was not a valid DNS server can be seen Appendix G.

Table 18. DHCPv6 Scopes

VLAN	Scope	Exclusion	Options 23 - DNS Servers
4	2001:db8:85a3:4::0/64	2001:db8:85a3:4::1	2001:db8:85a3:1::2
5	2001:db8:85a3:5::0/64	2001:db8:85a3:5::1	2001:db8:85a3:1::2

When testing DHCP by having Win7-1 attempt to grab an IPv6 address, the test failed. Recall this happened with IPv4 as well. The L3 switch needs to relay the DHCP requests. Researching the Cisco site, this reference proved valuable in this situation [25]. As the reference states, the command is slightly different in IPv6.

On VLAN interfaces 4 and 5 the following command were added:

Ipv6 dhcp relay destination 2001:db8:85a3:2::2

Now that there are relay agent commands on the switch, the '*ipconfig /renew*' command was used. This command never told the client to request an IPv6 address. Research shows that there is a new command switch for the *'ipconfig'* command. The new command is *'ipconfig /renew6'*. Once the scopes were ready, and the new command was executed the win7-1 client successfully got an IPv6 address. While tests were performed to verify access to other LANs' resources, such as the file server, it was discovered that the client could not communicate beyond the gateway. This makes sense since during the scope creation no Gateway or route scope option was available. In IPv4, the DHCP scope has an option for Router. A router option is not available in DHCPv6 scopes. During extensive troubleshooting and research of this routing issue it was finally determined the issue lies among the new features and options of DHCP.

IPv6 was designed to be dependent of DHCP for addressing. When a DHCP server is added to the environment it can compliment auto-configuration by adding functionality. The functionality it can add is handing out information beyond the minimal parameters needed to interact with a network, such as time servers, DNS servers and proxy information. Through the research of DHCPv6 options it was determined that for this network there were three ways to approach auto-addressing.

1. DHCP-less - Router advertises needed information. The only information that the router advertises is an IP, and route information. This is great on a simple network, but when other services are needed each node has to be manually configured for that service. For example, a network running with no DHCP server the DNS server IP would have to be manually entered into each workstation. This is acceptable in a small static environment.
2. Hybrid DHCP – This setup would have the router provide the IP addressing and routes, but additional information such as DNS servers would be handled by DHCP. This method is referred to as “stateless”. This method is advantageous for environments that do not care what the IP addresses of individual nodes are, and are accepting of the router using the Unique Identifier for the IPs.
3. Full DHCP – Also known as “stateful”, this scenario allows DHCP to give the IP addresses and all other information. The router still provides the routes. This method provides more control of the IP addressing. Using this method will allow an administrator to reserve IPs for certain services (like printers), exclude IPs from being handed out, and most of all, more control of what IPs are given.

For this network, simulating a mid-sized business network, the appropriate option would be a full DHCP implementation for the following reasons:

1. This network provides wireless. This assumes that systems will be added and removed from the network often and randomly. This would make the manual configuration aspect an administration hassle. This rules out the DHCP-less option
2. Although it is possible that the hybrid solution would be sufficient, having the extra control will potentially save time as the network grows. For the reason of not knowing what the future holds, the stateful configuration will be used.

Again, deploying the full DHCP setup means the client will get its IPv6 address and other settings from a DHCP server. In order to deploy the full-DHCP setup in this environment the router was configured to tell the clients to get certain information from a third party. This was accomplished by applying the following commands to the VLAN interfaces that will have DHCP clients attached, e.g. VLAN 4, 5.

ipv6 nd managed-config-flag – This command enabled the router to tell the clients to retrieve IP address from third party.

ipv6 nd other-config-flag – This command enables the router to tell the clients via a network-discovery message to get other options from a third party system.

The commands that were executed on the L3 switch are:

Conf t

int vlan x

ipv6 nd other-config-flag

ipv6 nd managed-config-flag

It is important to note if the router is not capable of this command set and upgrading the OS does not provide the features, these commands are available on each client. The problem with this is that the commands need to be run on each system and each time the computer boots. The commands are not persistent. On windows, doing a "*netsh*" command will give options to configure the client to look for a DHCP server for its information.

Once these commands were configured on the switch, the clients successfully request and receive DHCP information from the DHCP server (2008-2). The clients not only retrieved a DHCP IP address, they also received the DNS IP address. Pinging to the DNS server by IP was successful, but pinging by hostname failed. The inability to resolve hostnames gives focus back on the initial configuration of the DHCPv6 scopes. The error message stated that the IP was not a valid DNS server.

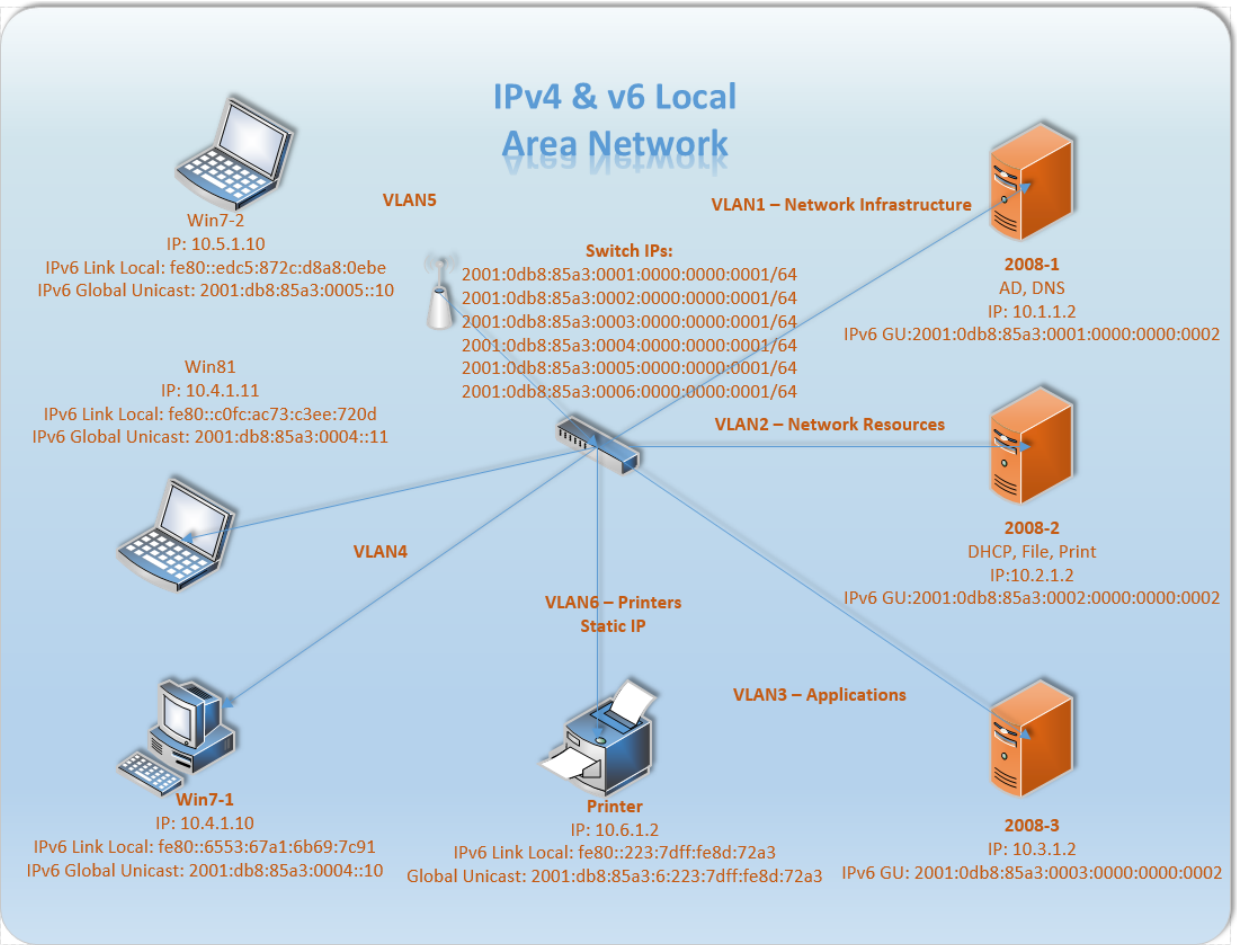
Troubleshooting this issue by reviewing the configuration on the DNS server brought attention to the DNS Listeners. Refer to the Appendix G for a few screenshots of the listener properties files. The listener configuration only listed IPv4 addresses of the computer. In this case, just the 10.1.1.2 IP address. This needed to reflect IPv6 addresses in order for a client to point to an IPv6 address. According to the an article on Network World, running the following command will resolve this issue [37].

Dnscmd /config /enableIPv6 1

The command was run on the 2008-1 and the DNS service was restarted. Second check of the DNS listener properties page showed the IPv6 addresses. This confirmed the IPv6 address was selected and saved in the settings. Retesting the IPv6 address on the DHCP scopes and the validation test succeeded. DNS queries from a client were successful as well. Now that all systems and services have been dual stacked, the second phase of testing was completed and all

use cases tested successfully, see Appendix E for the results. The current network topology looks like Diagram 5. The switch configuration up to this point can be found at Appendix H.

Diagram 5. IPv4/IPv6 LAN – Full Dual Stack Topology



4.4: Phase 3 - Remove IPv4

The state of the network at this point was a fully functioning IPv4/IPv6 network. At this point it was time to start disabling, deactivating and uninstalling IPv4 protocol(s) and services.

4.4.1: Remove IPv4 from Clients

Continuing the strategy of starting with the least impactful system(s) the wired Windows clients were the first to have IPv4 removed. To begin, a continuous ping was initiated by executing *ping 2008-2 –t* from the Operating System's command line utility. This ping is

executed to see if any interruption on the connection happens while disable IPv4. Proceeded to change the configuration on Win7-1, Win81, and Win7-2 to reflect IPv4 disable/uninstall. No interruption was experienced. The client computers now have all IPv4 settings removed, and the following tests were performed successfully:

- Printing
- Ping IPv6 address and DNS address
- File access
- Web browse

A final test was run and that was pinging an IPv4 address. As predicted, this test was unsuccessful.

4.4.2: Remove IPv4 from Printer

Next node on the network to have IPv4 removed was the printer. From one of client workstations, the printer was access by an Internet browser using the IPv6 address. When accessing an IPv6 address using http the URL needs a slight modification. Once access is gained using the IPv6 URL is http://[2001:0db8:85a3:0003::2], the IPv4 settings were unchecked, in essence disabling the protocol. Test pages were sent to the printer and as suspected the pages did not print and the printer was in an offline state. This was because the printer's server was pointing to the IPv4 address. After logging on to the print server and adding an IPv6 port, the printer was changed to point to the new port. Once the apply button was pressed the queued print jobs printed. The printer was now back online via IPv6-only. A huge benefit of using a print server, as was done on this network, is that changes only need to be performed in one location. No client pointing to the printer via a Print server will need to have anything changed.

This can be a significant benefit to a large organization as it will save large amount of configuration time.

4.4.3: Remove IPv4 from the Servers

The next devices were the servers; starting with 2008-3, the web server. Same as the Win7 clients, the IPv4 protocol was removed. A few tests were successfully run on the server to make sure it could still communicate with the network. Clients successfully accessed the web page hosted by 2008-3. The 2008-2 server (DHCP, file, and printer server) was next and the protocol was successfully removed.

The final Windows system was the Domain Controller and DNS server. Once the protocol was removed from the system, a user account's password was changed and an attempt to logon from a client using the account was successful. This test proves the client of the domain was not using cached credentials and that the client was truly communicating with the domain controller to confirm credentials were accurate.

4.4.4: Remove IPv4 from the L3 Switch

Now that all IPv4 is removed from all nodes on the network, the L3 switch had all the IPv4 settings removed. The first configuration removed was the IPv4 routing.

No ip routing

Next configuration removed from the switch was the IPv4 IP addresses assigned to the VLAN interfaces, and for the VLAN interfaces with DHCP relay, those commands were removed as well.

Int vlan x

no ip address 10x.1.1 255.255.255.0

no dhcp-relay 10.2.1.2

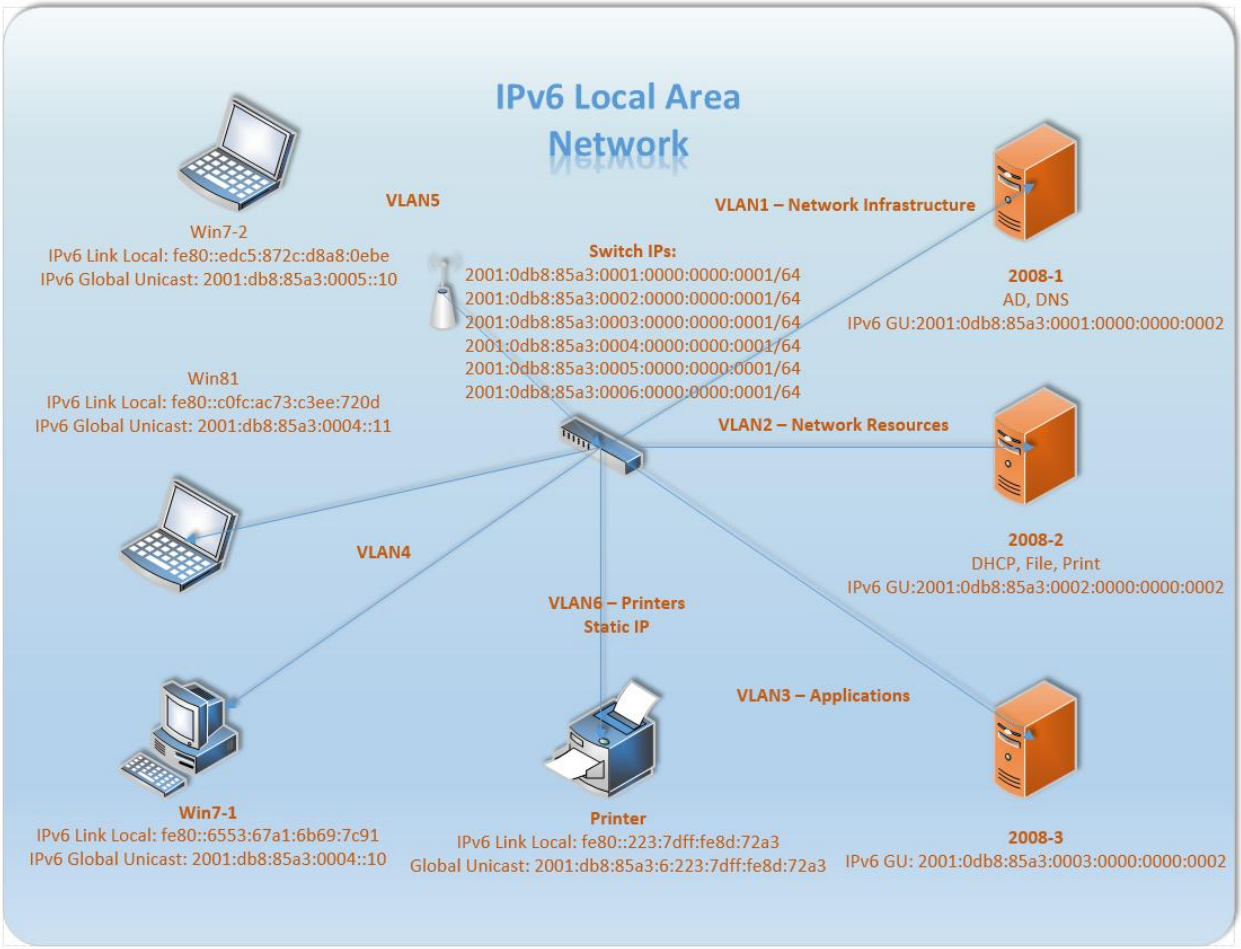
The Linksys WAP200 does not support IPv6 for managing. In order to manage this device a client will have to be configured with an IPv4 address and placed on the same subnet as the AP. At this time the IPv4 client can access the AP via the IPv4 address.

4.4.5: Remove IPv4 from Services

Now that all IPv4 traffic is gone, the services supporting IPv4 were updated. Within DNS there were only a few more IPv4 records remaining. The ones remaining were the Server IPs. They were left due to they were static IPs. The clients IPs were removed when the IPv4 protocol was removed from the system. Selecting the IPv4 addresses and deleting was all that was needed on the DNS server. DHCP still had the IPv4 scopes. Those were easily deleted as well.

At this point, with the exception of the Access Point, the network was solely IPv6. Wireshark on the SPAN port of the L3 switch confirmed no IPv4 traffic. The final test phase was conducted and all but the access to the Access Point passed. See Appendix E for the test results. As of now the network looked like Diagram 6. The completed L3 switch configuration can be found in Appendix I.

Diagram 6. IPv6 LAN – IPv6 Only Topology



Chapter 5: Discussion

In the previous section, a case study was presented in which an IPv4 network was built and converted to an IPv6 network. The project consisted of three phases: Phase 1 – building the IPv4 network, Phase 2 - installing IPv6 leveraging dual stack and edge-in, and Phase 3 - removing IPv4. In this Section, the results of each phase are discussed including challenges that occurred during the conversion.

5.1: Phase 1 - IPv4 Network Build

The first phase was to build an IPv4 network. The network consisted of nodes and services that are typically found on a SMB network. Nodes included devices such as servers, clients, switches, access points, printers. Services included resources such as web pages, files sharing, DNS, and active directory. When the network was complete, a predetermined set of use case scenarios were tested. These use cases were items like a user logons on to a client and prints a file accessed via a file server, or a laptop is connected to the network via a wireless signal and gets an IP address from a DHCP server (see Appendix J for a complete list of use cases). Once all the test cases were performed successfully, the conversion process began.

5.2: Phase 2 - Dual Stack Implementation

Phase 2 involved implementing a dual stack environment (i.e. two protocols on each node). This involves installing the IPv6 protocol on all nodes of the network and enhanced the services to support the IPv6 protocol. The strategy used to this phase was an ‘edge-in’ approach, essentially starting from a device with little impact on the network and working my way toward more critical systems.

When the IPv6 protocol was enabled on the devices, the device auto generated an IPv6 address. The IPv6 address that was generated was a link local IPv6 address. As proven from the experiment, these addresses are unable to communicate with systems on other networks presenting the first challenge during the conversion process.

5.2.1: Anticipated Challenge – IP Address Scheme

The first predicted challenge of this project was deciding what type of IPs to use for this network. In order to get systems communicating with systems on other networks an IP address scheme needed to be selected. This decision is based on the size and type of network. This network was simulating a corporate network and it would more than likely need to communicate with the Internet at some point. Since the network would need access to other networks there were two options. The first option was Unique Local addresses (ULA), comparable to Private IPs. Using ULAs would require a system to convert IPs to and from Unique and Global. The second option was to use Global Unique (GUA) address scheme. GUA Addresses are comparable to Public Internet IP addresses. A benefit to IPv6 is the essentially unlimited amount of IPs available. With the massive amount of IPs it is not as critical to conserve IP addresses as is required with IPv4. The only benefit to using ULAs would be the false sense of security of having a different IP address when accessing internal network; which actually is not only false security, it is also more complicated and adds a single point of failure via the NAT device. Considering this information, the correct option was to go with a Global Unique Address space for the whole network.

5.2.2: Unanticipated Challenge - Software Compatibility

During phase two of the experiment a few unexpected issues were encountered. The first unpredicted challenge of this process was configuring the switch to support IPv6. Cisco has had support for IPv6 for quite some time, but it was evident from this project that their equipment may not come with all the features installed and ready to use. This became evident when trying to configure IPv6 addresses on the VLAN interfaces of the switch. When receiving an error when assigning an IPv6 address to the switch it came apparent that the switch template needed to be changed. Once the template issue was resolved, another issue with the switch surfaced. The second issue with the switch was once the ability to configure IPv6 addresses was performed, being able to route to and from networks was not possible. When trying to activate IPv6 routing it was concluded that the IOS the switch was running was capable of IPv4 routing but not IPv6 routing. To resolve this issue the IOS needed to be upgraded from an IPBASE image to an IPSERVICE image. This is odd considering the IPBASE image is capable of IPv4 routing. Logic would have thought that if the switch was able to route on IPv4, it would be able to route on IPv6; clearly not the case.

5.2.4: Unanticipated Challenge - Hardware Incompatibility

The second unpredicted challenge was involving the Linksys WAP200 Business Series Access Point. Knowing the Linksys product line is now owned by Cisco, and the age on this particular device is less than 5 years old, one would have thought the access point was capable of being managed by an IPv6 IP. During the migration of this device it was realized that this was not the case. Upgrading the firmware was ineffective as well.

5.2.5: Anticipated Challenge - IP Addressing - DHCP

The next anticipated challenge encountered during this phase dealt with the DHCP services. Upon configuring the switch to relay DHCPv6 requests, a DHCP client was still unable to get a client to grab an IP address from a DHCP server. Research showed that was designed to not need DHCP in order to function. DHCP is primarily used to give IP addresses to clients. IPv6 has built-in mechanisms to eliminate the need for DHCP. A system accessing an IPv6 network for the first time can get all the needed information from the information advertised by the router in order to connect to a new network; therefore DHCP is not needed. This was evident from the experiment because before introducing a DHCP server to the IPv6 network, all the devices that were either configured to automatically configure IP (clients) or were statically set (servers) were able to communicate with all aspects of the network.

During further research, it was learned that DHCP can be used with IPv6 to add features and capabilities, essentially complimenting IPv6 auto configuration mechanism. The way DHCPv6 can compliment a network is by providing additional parameters for the clients to automatically obtain. Additional parameters can include settings such as DNS server IP, and time servers. If a DHCP server was not present on a network, it is most certain that the additional parameters would need to be manually configured on each client automatically obtaining IP settings. Knowing this network is simulating a midsized business network, it was obvious that the clients would need a DNS server automatically provided to them, yielding the need for DHCP in some capacity.

DHCP could be involved in two ways. One way would be fully integrated. The fully integrated method, also known as stateful, would mean that the DHCP server would provide all parameters with the exception of the gateway/router IP address. The second method would be a partially integrated solution. This solution, known as stateless, would allow the client to auto

generate it's IP from the information provided by the router. During both methods, a client would get the gateway from the router. Using the stateful method will allow Network Administrators more control on the IPs given to the clients; allowing them to use self designed IP scheme for the host portion of the IPs. To give me more control of the IPs given, the stateful configuration was selected.

In order to enable DHCPv6's interaction with the network, the switch needed to be configured to tell the client to obtain certain information from a 3rd party DHCP server; essentially telling the client to multicast a request for more information. This was done by using two commands. One command tells the router to tell clients to get other information from a DHCP server. The other command tells the router to tell the clients to get its IP address from a DHCP server. Once configured, the clients were able to obtain an IP address and DNS server IP from the DHCP server.

5.2.6: Anticipated Challenge - DNS

The final anticipated challenge was with DNS. It was anticipated that DNSv6 would require a modification, upgrade or reinstall of the service. Surprisingly, this was not the case. In fact, once IPv6 was enabled on a Windows device, that device communicated its IPv6 address to the DNS server. The DNS server, whether it had an IPv6 address or not, updated the table to include the AAAA record. DNS services on Windows 2008 include full support of IPv6

5.2.7: Conclusion

At this point in the experiment all nodes and services with the exception to the Access Point were configured for IPv6. After completing phase two of the test plan, the third phase of the experiment was completed. See Appendix E for the test plan.

5.3: Phase 3 - IPv4 Removal

Phase 3 encompassed the removal of IPv4. Once again, a less impactful approach was carried out by starting with the clients, moved on to the printer, and then the servers. Once IPv4 was completely removed from the nodes, all IPv4 settings were removed from the switch. Services were modified and cleaned up. The DHCP service was still providing IPv4 addresses for clients that might request an IPv4 address; therefore the IPv4 scopes were removed. The final few 'A' records were deleted from DNS. As predicted, phase three was uneventful and went smoothly.

Chapter 6: Conclusion

Overall the conversion was a success. The Cisco switch required some advanced configurations, but Cisco has made IPv6 compatibility available. Initial research of Cisco's IOS, and other features would be beneficial to avoid delays and surprises. Microsoft's supported operating systems are prepared and ready for IPv6. In fact, the supported Operating Systems are IPv6 enabled "out-of-the-box". The Microsoft DNS issue that occurred, the IPv6 address was not available as a "listener", was a bit of an overlook on Microsoft's part, but Microsoft had lots of white papers, and blogs on their websites. A service pack probably addresses this issue.

The only network node unable to convert to full IPv6 support was the Linksys Access Point. The access point is available for use by wireless clients; however managing the device is more difficult. Replacing this device would be the best solution to this issue, but a low cost work around to this issue could be use of a "management" workstation to access the device. The "management" workstation would be a system running a dual stack configuration, and would reside on the same IPv4 subnet as the Access Point. When the AP needs managed, an administrator would use the IPv6 IP to access the "management" system, and bounce to the AP's IPv4 address; in essence, bridging the two separate networks via the multihomed system. Even though this Access Point was a business class device, Linksys devices are behind on IPv6 readiness.

6.1: Lessons Learned and Business Considerations

Considering the unexpected AP issue and the need to do multiple ad hoc software modifications on the Cisco L3 switch, a key lesson learned was the need to do an evaluation of the network equipment before starting the conversion. The evaluation would include hardware and software IPv6 compatibility investigations. Although not a show stopper to this project,

definitely something that a business should do in order to avoid delays and unplanned costs. Other considerations a business may want to consider before converting to the IPv6 protocol are:

- What are the benefits to the business?
- Training of technical IT staff
- Setting up a lab

6.2: Suggested Project Plan

As with the majority of large scale IT Projects, planning is the key to success. Below is a suggested approach to an IPv6 conversion project. This approach can be used as a starting point for businesses who are considering the move from IPv4 to IPv6.

A company's approach to an IPv6 conversion project might look something like this:

1. **Establish a Project Team** – The team should include a member from each IT entity, i.e. Applications team, infrastructure, Network, etc… Perhaps procure an IPv6 consultant for expertise.
2. **Evaluate Business Needs** – How is this beneficial to the company? The consultant could help here. Hardware Inventory should happen during this step, to determine extra costs. Hardware Inventory:
    1. Determine what is IPv6 compatible
    2. Determine what will need replaced and what can be upgraded
3. **Get Stakeholder buy-in** – the benefits from the evaluation of business needs will be key for getting Stakeholder buy-in.
4. **Training of Technical IT personnel** – Important to have this done before the project to avoid unplanned delays.

5. **Contact and involve ISP throughout the planning**
    a. Obtain IP address space
    b. Confirm the ISP readiness
6. **Setup a Lab simulating the Existing network**
    a. Lab should include same hardware, and software levels
    b. Lab should include corporate applications
7. **Testing**
    a. IT testing
    b. User Testing
8. **Design an Implementation Plan** - This should include an upgrade path for non compatible hardware/software. Have phases this will break the project up to allow for milestones check points.
9. **Start the Implementation** – Communicate well throughout the business, remind key management of the value, and why this is being done.

6.3: Future Work

A suggested next step to this research would be connecting the IPv6 network to the Internet. This would require involvement with an ISP. When contacting the ISP it would be important to know if the ISP is IPv6 ready.

If the ISP is IPv6 ready, the ISP would allocate an IPv6 Global Unicast Address space. Since this research was isolated from the Internet, and a random IPv6 Global Unicast address space was selected, a potential challenge to this path would be the likelihood of having to re-IP the network to the IP scheme the ISP provides. This would be a great opportunity to evaluate the theory that IPv6 methodology allows for easy re-addressing. A benefit to IPv6's approach to

being able to re-IP easily would prevent a company from being “locked in” to a service provider. Performing the re-IP would give a proof of concept and ease the anxiety of a company considering an ISP change.

Another potential challenge to this course would be addressing the issue of having this IPv6 network interact with legacy IPv4 networks. Most likely having to incorporate one or more of the transition mechanisms discussed in this research. For example, in order for an IPv6 network to communicate with an IPv4 network one might consider using Protocol Address Translation (NAT-PT). NAT-PT would translate IPv6 to IPv4.

If the ISP is not IPv6 ready, then an alternate approach would be required; yielding different challenges, such as having to tunnel IPv6 over the IPv4 network of the ISP. This would give the opportunity to evaluate the tunneling transition mechanism. In this scenario tunneling is encapsulating the IPv6 network datagrams in an IPv4 datagram. The Internet edge device would need to encapsulate the traffic that is destined for the Internet. Collaboration with the far end may need to be taken into consideration.

References

1. Marsan, C.D., *At long last, Obama highlights IPv6 issue*, in *Network World*2010.
2. *Vint Cerf*. 2/21/2014]; Available from: http://en.wikipedia.org/wiki/Vint_Cerf.
3. McNamara, P., *Why IPv6? Vint Cerf keeps blaming himself*, in *Network World*2010: Internet.
4. S. Deering, R.H. *Internet Protocol, Version 6 (IPv6)*. RFC 1998; Available from: http://tools.ietf.org/html/rfc2460.
5. P. Srisuresh, M.H. *IP Network Address Translator (NAT) Terminology and Considerations*. 1999; Available from: http://tools.ietf.org/html/rfc2663.
6. Riley, D.J., *Adopting IPv6 is a Corporate Business Issue*, in *Computer World UK*2013.
7. Program, D.I. *Internet Protocol*. RFC 1981; Available from: http://tools.ietf.org/html/rfc791.
8. A. Marine, N.N. *FYI on Questions and Answers to Commonly asked "New Internet User" Questions*. 1994; Available from: http://www.ietf.org/rfc/rfc1594.
9. Hagen, S., *IPv6 essentials*, 2006, O'Reilly: Farnham.
10. J. Mogul, S.D. *Path MTU Discovery*. RFC 1990; Available from: http://tools.ietf.org/html/rfc1191.
11. J. McCann, S.D., J. Mogul. *Path MTU Discovery for IP version 6*. RFC 1996; Available from: http://tools.ietf.org/html/rfc1981.
12. Matthew Luckie, B.S., *Measuring path MTU discovery behaviour*, in *IMC '10 Internet Measurement Conference*, M. Allman, Editor 2010: Melbourne, Australia. p. 109-122.
13. Nobile, A.-L. *ARIN IPv4 Countdown Plan*. 2012; Available from: https://www.arin.net/participate/meetings/reports/ARIN_XXIX/PDF/monday/nobile_ipv4_countdown.pdf.
14. Vaughan-Nichols, S.J. *The Internet is running out of IPv4 gas*. 2010.
15. Y. Rekhter, B.M., D. Karrenberg, E. Lear, G. J. de Groot. *Address Allocation for Private Networks*. RFC 1996; Available from: http://tools.ietf.org/html/rfc1918.
16. Technologies, C.f.N.G.
17. R. Droms, E., J. Bound, B. Volz, C. Perkins, M. Carney. *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*. 2003; Available from: http://www.ietf.org/rfc/rfc3315.
18. S. Cheshire, B.A., E. Guttman. *Dynamic Configuration of IPv4 Link-Local Addresses*. RFC 2005; Available from: http://tools.ietf.org/html/rfc3927.
19. S. Thomson, T.N. *IPv6 Stateless Address Autoconfiguration*. RFC 1998; Available from: http://tools.ietf.org/html/rfc2462.
20. Mockapetris, P., *DOMAIN NAMES - IMPLEMENTATION and SPECIFICATION*. 1983.
21. Marshall Brain, S.C. *How Domain Name Servers Work*. 2000; Available from: http://www.howstuffworks.com/dns.htm.
22. Huston, G. *IPv4 Address Report*. Available from: http://www.potaroo.net/tools/ipv4/index.html.
23. ARIN, *ARIN - Deploying IPv6*.
24. Greiner, L., *What If the Internet Went Down...and Didn't Come Back Up?*, in *CIO*January 15, 2008.
25. Cisco *Cisco - What Enterprises Should Do About IPv6 In 2010*. 2010.
26. Marsan, C.D., *No business case for IPv6, survey finds*, in *Network World*2009.
27. E. Nordmark, R.G. *Basic Transition Mechanisms for IPv6 Hosts and Routers* 2005; Available from: http://tools.ietf.org/html/rfc4213.

28. Grossetete, M.T.a.P., *IPv6 Integration and Coexistence Strategies for Next-Generation Networks*, in *IEEE Communications Magazine*2004.
29. Microsoft. *Microsoft Product Lifecycle Search - XP*. 2014; Available from: http://support.microsoft.com/lifecycle/search/default.aspx?sort=PN&alpha=windows+xp&Filter=FilterNO.
30. Microsoft. *Microsoft Product Lifecycle Search - 2003*. 2014; Available from: http://support.microsoft.com/lifecycle/search/default.aspx?alpha=Windows+Server+2003+R2.
31. Davies, J. *Support for IPv6 in Windows Server 2008 R2 and Windows 7*. 2014; Available from: http://technet.microsoft.com/en-us/magazine/2009.07.cableguy.aspx.
32. Blanchet, M., *Migrating to IPv6*2006.
33. Horley, E. *IPv6 Unique Local Address or ULA - what are they and why you shouldn't use them*. 2013; Available from: http://www.howfunky.com/2013/09/ipv6-unique-local-address-or-ula-what.html.
34. *Where is IPv6 on my Cisco 3560 in my CCNA lab?* 2013; Available from: http://www.certificationkits.com/blog/index.php?option=com_wordpress&p=436&Itemid=1.
35. Ritter, H. *ipv6 unicast-routing not available*. 2009; Available from: https://supportforums.cisco.com/discussion/10551686/ipv6-unicast-routing-not-available.
36. Morimoto, R. *IPv6 Static Addressing and DNSv6*. 2011; Available from: http://www.networkworld.com/community/blog/ipv6-static-addressing-and-dnsv6.
37. Microsoft. *IPv6 configuration items*. 2005; Available from: http://technet.microsoft.com/en-us/library/cc783049(v=WS.10).aspx.

Appendix

Appendix A – Switch Configurations

Appendix A – Sh Ver Command

```
IPv4IPv6Lab(config-if)#do sh ver
Cisco IOS Software, C3560 Software (C3560-IPBASE-M), Version 12.2(35)SE5, RELEASE
SOFTWARE (fc1)
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 19-Jul-07 18:15 by nachen
Image text-base: 0x00003000, data-base: 0x01100000

ROM: Bootstrap program is C3560 boot loader
BOOTLDR: C3560 Boot Loader (C3560-HBOOT-M) Version 12.2(35r)SE2, RELEASE
SOFTWARE (fc1)

IPv4IPv6Lab uptime is 1 week, 6 days, 20 hours, 33 minutes
System returned to ROM by power-on
System image file is "flash:c3560-ipbase-mz.122-35.SE5/c3560-ipbase-mz.122-35.SE5.bin"

cisco WS-C3560-8PC (PowerPC405) processor (revision A0) with 122880K/8184K bytes of
memory.
Processor board ID FOC1148ZBMD
Last reset from power-on
6 Virtual Ethernet interfaces
8 FastEthernet interfaces
1 Gigabit Ethernet interface
The password-recovery mechanism is enabled.

512K bytes of flash-simulated non-volatile configuration memory.
Base ethernet MAC Address       : 00:1E:BD:9C:FA:00
Motherboard assembly number     : 73-10612-07
Power supply part number        : 341-0207-01
Motherboard serial number       : FOC11482J8B
Power supply serial number      : LIT1145054X
Model revision number           : A0
Motherboard revision number     : C0
Model number                    : WS-C3560-8PC-S
System serial number            : FOC1148ZBMD
Top Assembly Part Number        : 800-28131-01
Top Assembly Revision Number    : C0
Version ID                      : V01
CLEI Code Number                : COM8C00ARA
Hardware Board Revision Number  : 0x01


Switch   Ports  Model              SW Version            SW Image
------   -----  -----              ----------            ----------
*    1   9      WS-C3560-8PC       12.2(35)SE5           C3560-IPBASE-M
```

Appendix A – Switch Running Configuration

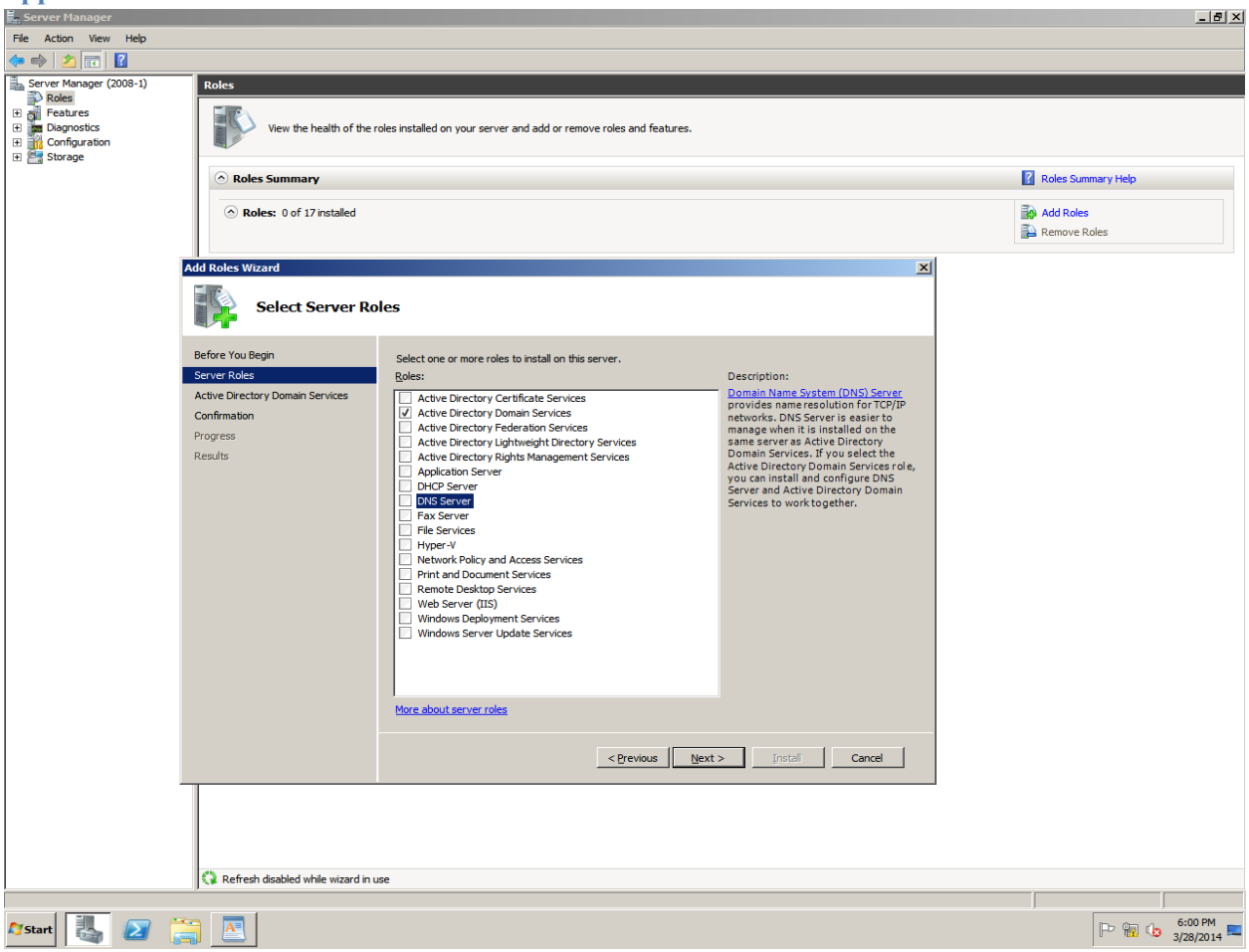
```
Current configuration : 1727 bytes
!
version 12.2
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname IPv4IPv6Lab
!
!
no aaa new-model
system mtu routing 1500
ip subnet-zero
ip routing
!
!
enable password cisco
!
!
no file verify auto
spanning-tree mode pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
!
interface FastEthernet0/1
 description server-1
 spanning-tree portfast
!
interface FastEthernet0/2
 description server-2
 switchport access vlan 2
 spanning-tree portfast
!
interface FastEthernet0/3
 description server-3
 switchport access vlan 3
 spanning-tree portfast
!
interface FastEthernet0/4
 description Access Point
 switchport access vlan 4
 spanning-tree portfast
!
```

```
interface FastEthernet0/5
 description laptop
 switchport access vlan 5
 spanning-tree portfast
!
interface FastEthernet0/6
 description Printer
 switchport access vlan 6
 spanning-tree portfast
!
interface FastEthernet0/7
 description laptop win7
 switchport access vlan 5
 spanning-tree portfast
!
interface FastEthernet0/8
 description SPAN port
 shutdown
!
interface GigabitEthernet0/1
 switchport trunk encapsulation dot1q
 switchport mode trunk
 shutdown
!
interface Vlan1
 ip address 10.1.1.1 255.255.255.0
!
interface Vlan2
 ip address 10.2.1.1 255.255.255.0
!
interface Vlan3
 ip address 10.3.1.1 255.255.255.0
!
interface Vlan4
 ip address 10.4.1.1 255.255.255.0
 ip helper-address 10.2.1.2
!
interface Vlan5
 ip address 10.5.1.1 255.255.255.0
 ip helper-address 10.2.1.2
!
interface Vlan6
 ip address 10.6.1.1 255.255.255.0
!
ip classless
ip http server
```

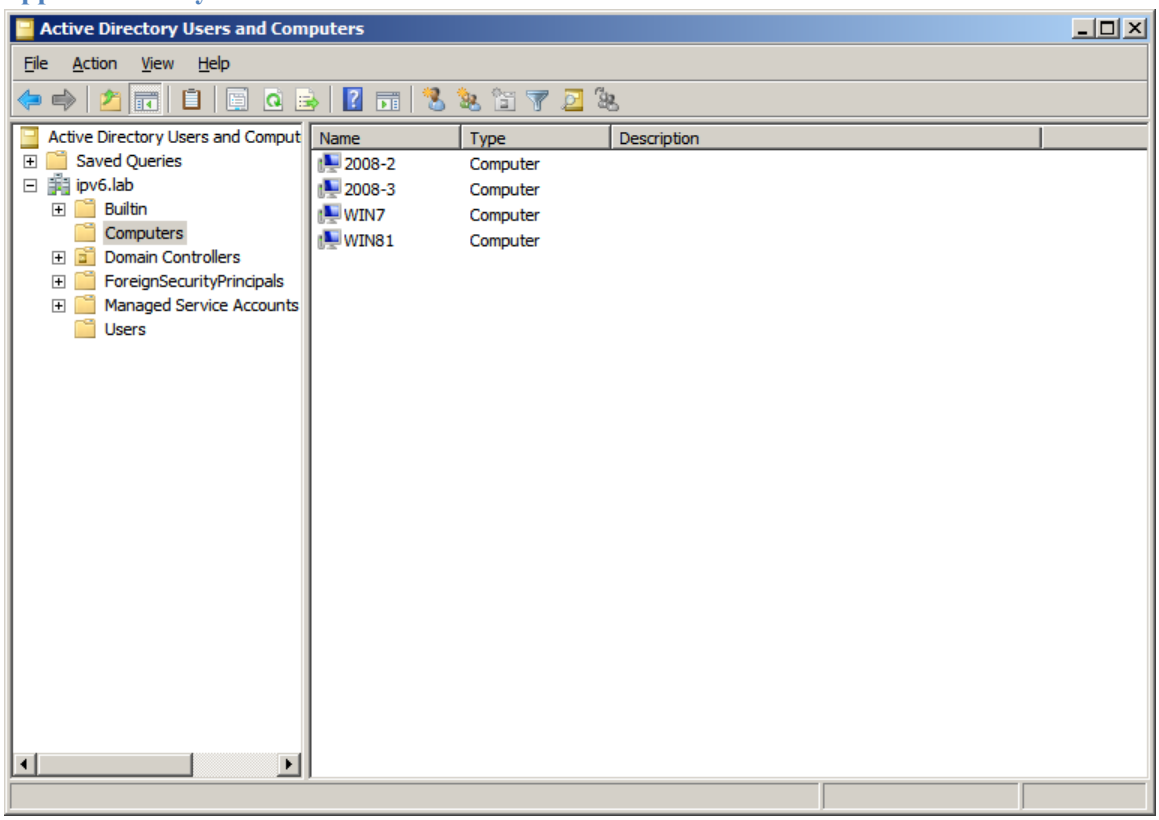
```
!
!
control-plane
!
!
line con 0
line vty 0 4
 exec-timeout 0 0
 password cisco
 login
line vty 5 15
 password cisco
 login
!
!
monitor session 1 source vlan 1 - 6
monitor session 1 destination interface Fa0/8
end

IPv4IPv6Lab#
```

Appendix B – 2008-1 Screenshots

Appendix B – AD/DNS Install

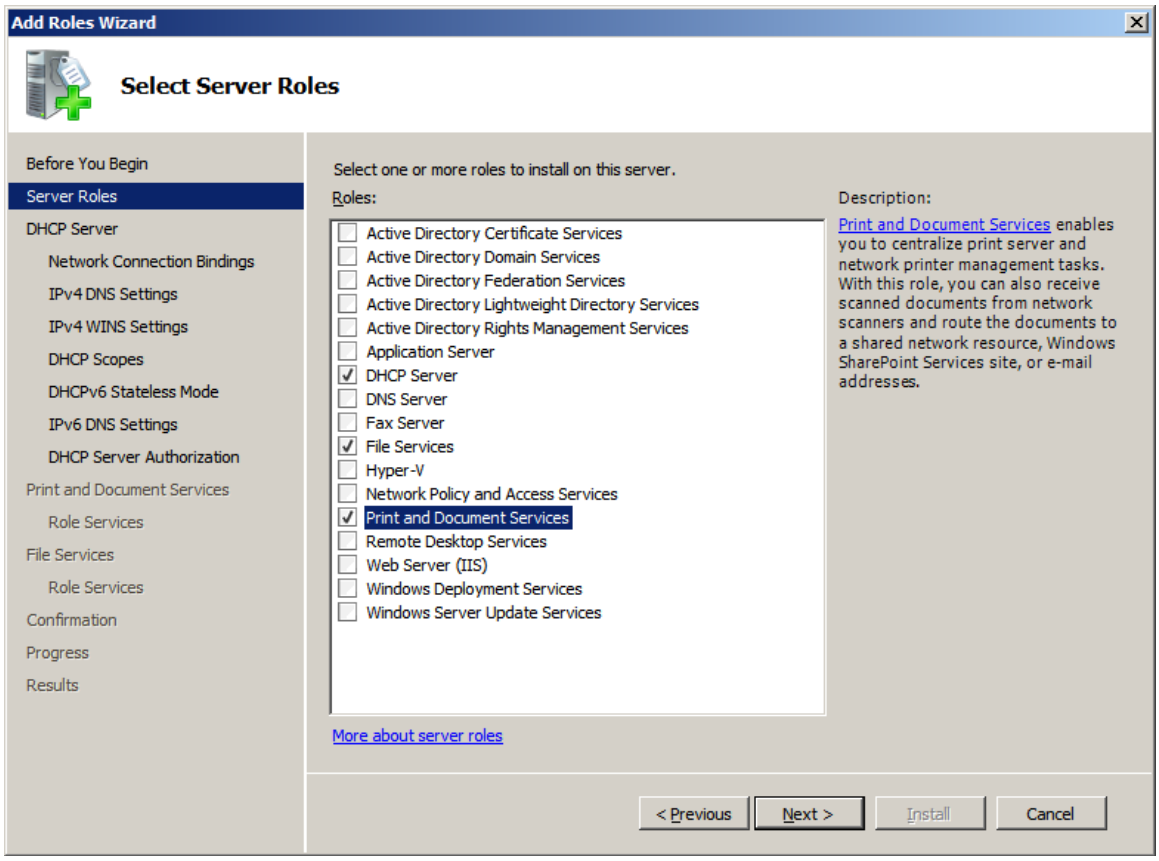


Appendix B – Systems on Domain

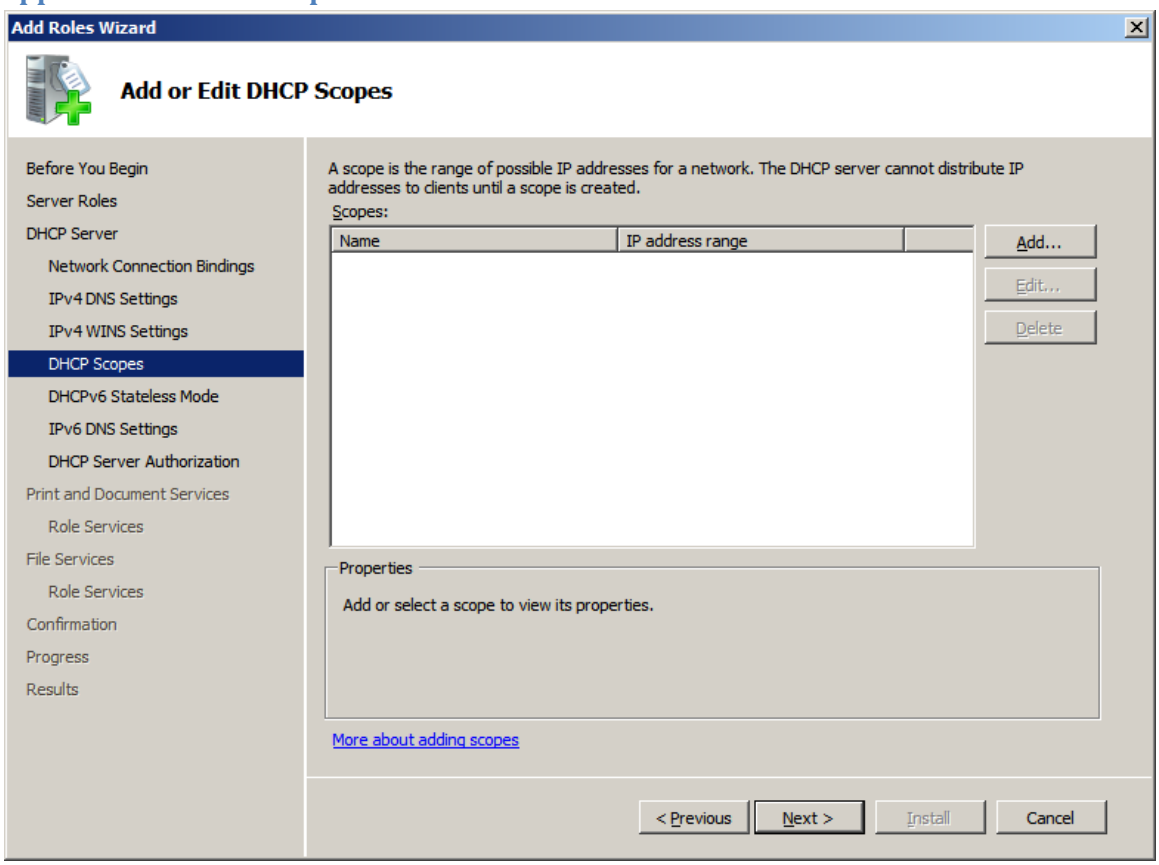


Appendix C – 2008-2 Screenshots

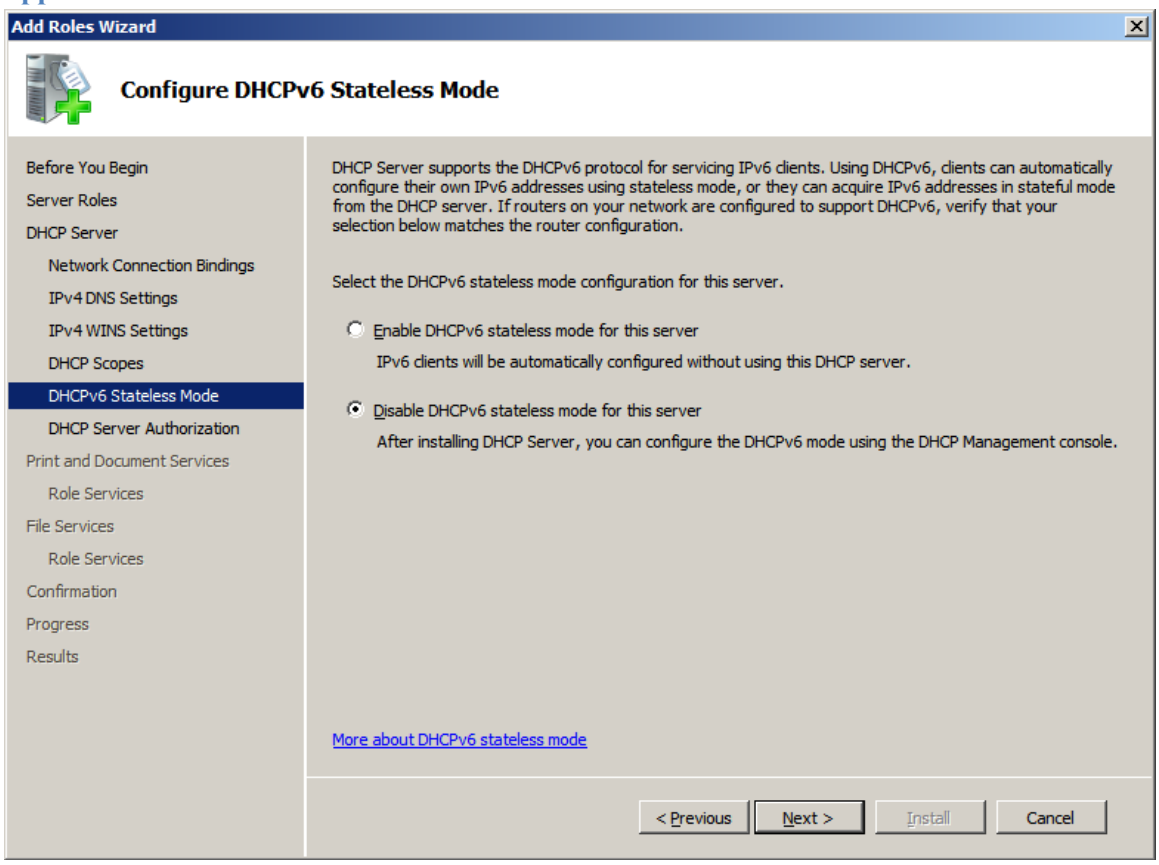
Appendix C – DHCP/File/Print Services Install



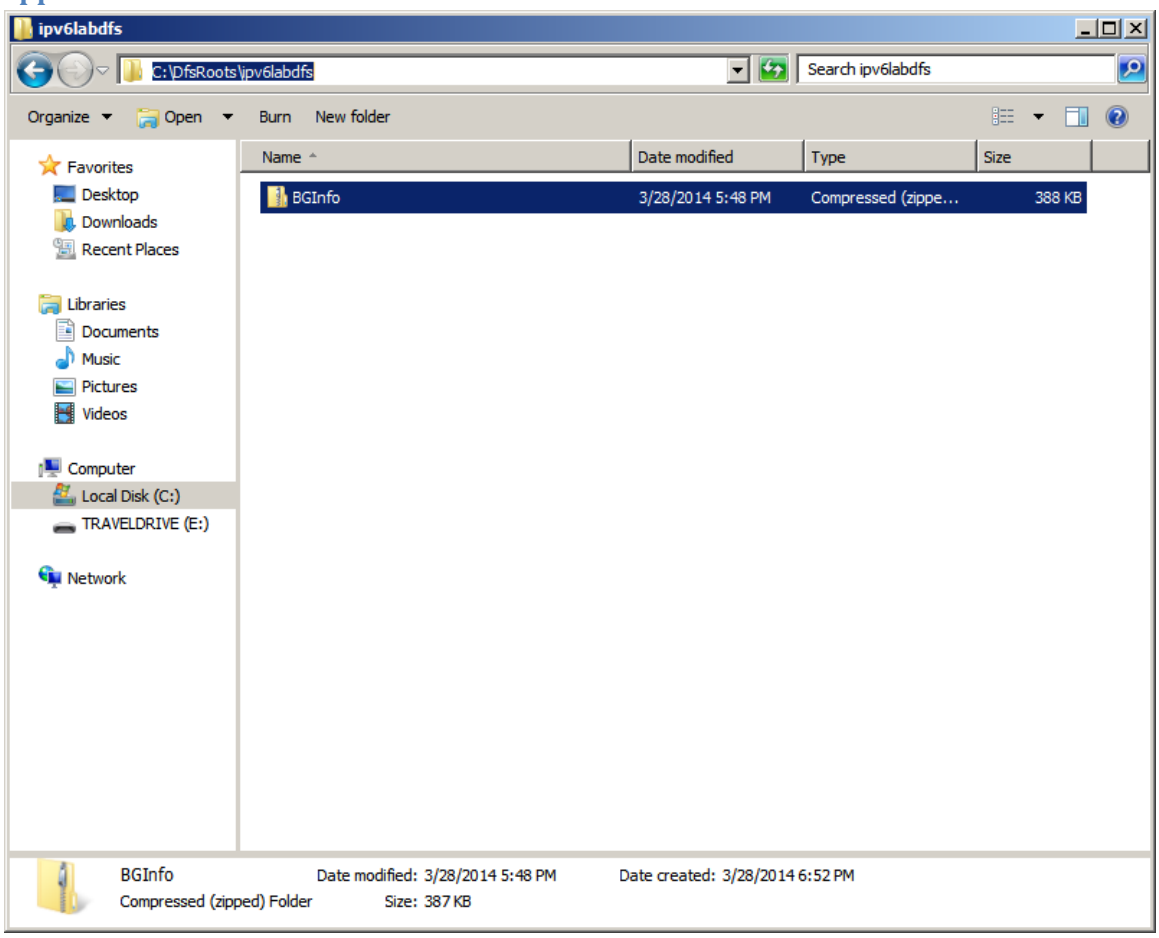
Appendix C – DHCP Scopes



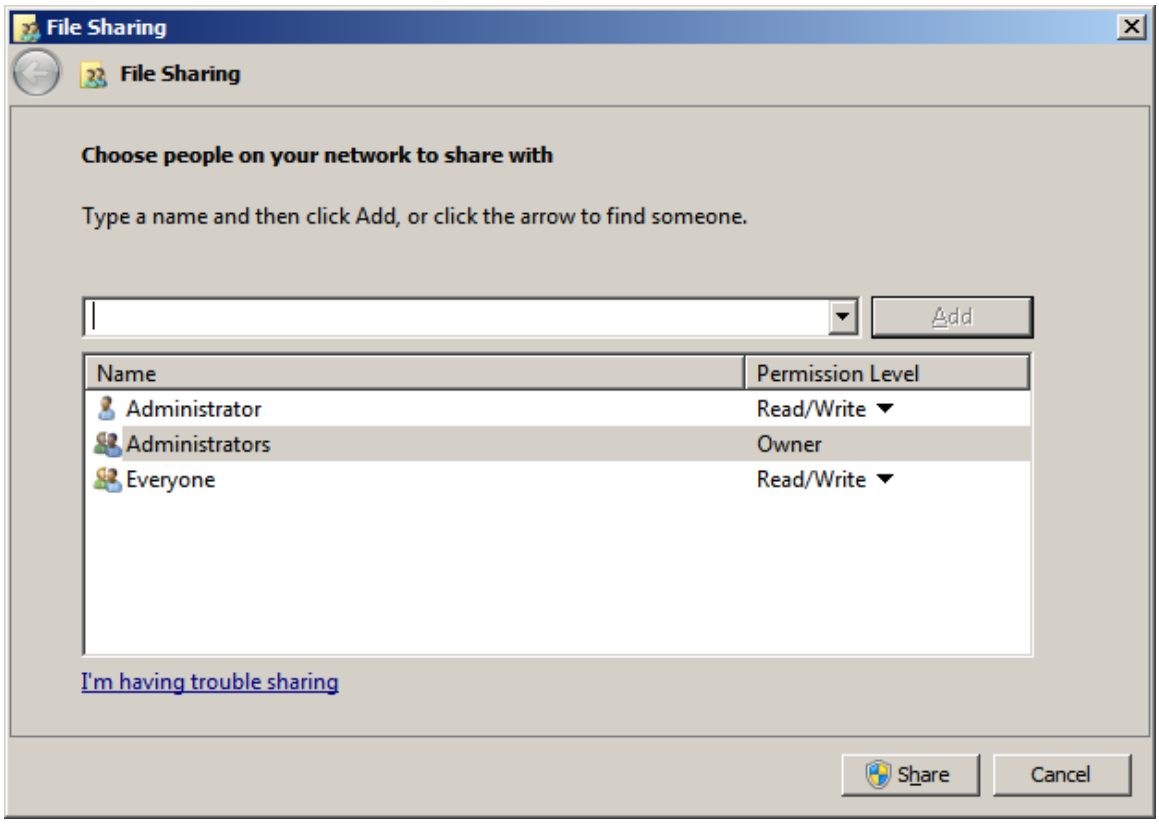
Appendix C – DHCP Stateless Mode



Appendix C – File Share



Appendix C – File Permissions on Share



Appendix D – 2008-3 Screenshots

Appendix D – II7 Test

Appendix E – Test Plans

Appendix E – Test Phases

3 test phases:

1. At end of IPv4 network build
2. At end of Dual stack and network services upgrade
3. At end of IPv4 removal

At these phases the following will be tested:

Workstation/Server test cases:

- Connect to wireless
- DHCP
- DNS
- Join Domain

User test cases:

- Logon/off
- Print
- Web Browse
- Access Files

Administrator test cases:

- Manage Domain (RDP to servers)
- Manage Access Point
- Manage Switch

Appendix E – Phase 1 Test Plan

Test Plan			
Phase:	1	Success	Fail
User	Logon/off (w new account)	x	
	Print (includes adding printer)	x	
	Web Browse	x	
	File Access	x	
Workstation/Server	Connect wireless	x	
	DHCP	x	
	DNS	x	
	Join Domain	x	
Administrator	Manage Domain (RDP to servers)	x	
	Manage AP	x	
	Manage Switch	x	

Appendix E – Phase 2 Test Plan

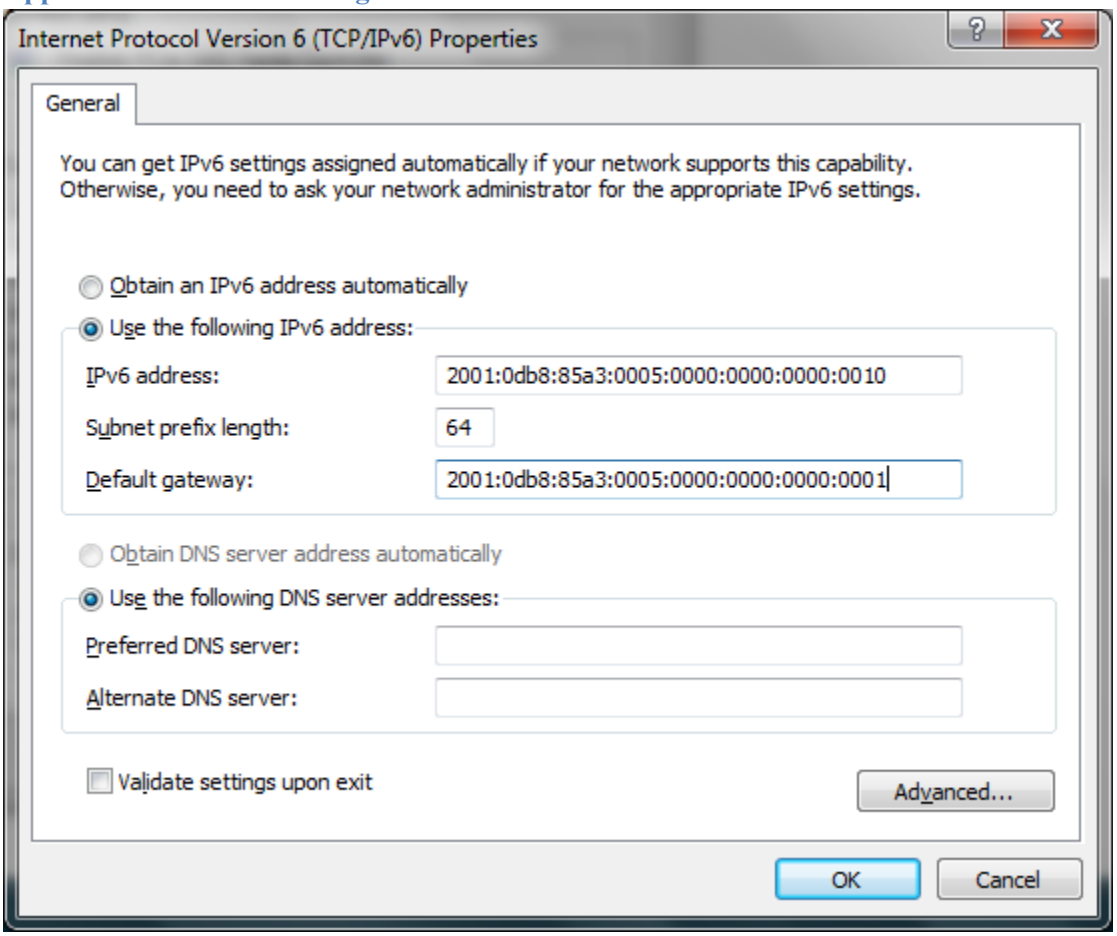
Test Plan			
Phase:	2	Success	Fail
User	Logon/off (w new account)	x	
	Print (includes adding printer)	x	
	Web Browse	x	
	File Access	x	
Workstation/Server	Connect wireless	x	
	DHCP	x	
	DNS	x	
	Join Domain	x	
Administrator	Manage Domain (RDP to servers)	x	
	Manage AP	x (via IPv4)	
	Manage Switch	x	

Appendix E – Phase 3 Test Plan

Test Plan			
Phase:	3	Success	Fail
User	Logon/off (w new account)	x	
	Print (includes adding printer)	x	
	Web Browse	x	
	File Access	x	
Workstation/Server	Connect wireless	x	
	DHCP	x	
	DNS	x	
	Join Domain	x	
Administrator	Manage Domain (RDP to servers)	x	
	Manage AP		x
	Manage Switch	x	

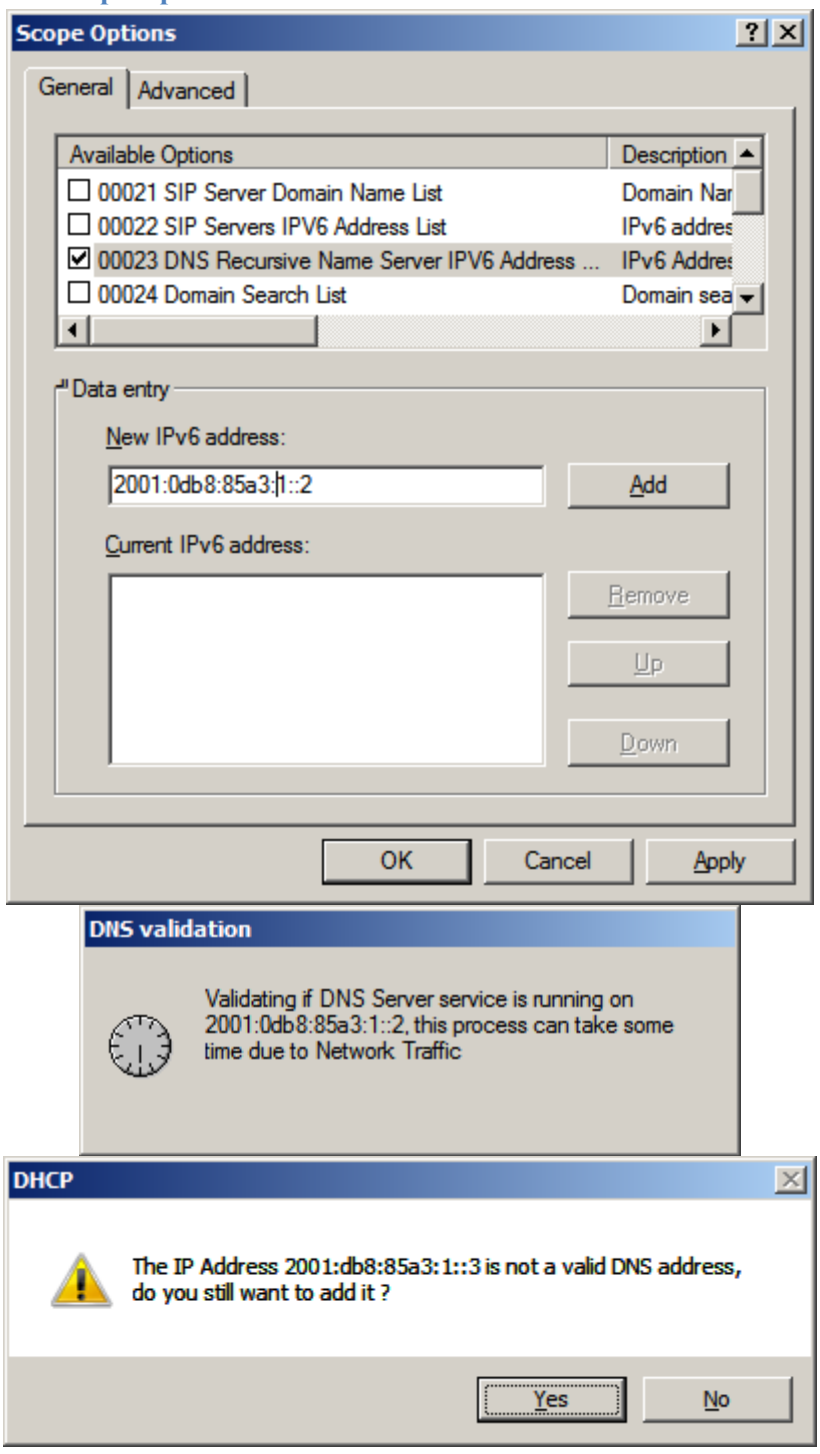
Appendix F – IPv6 Manual Configuration

Appendix F – Manual Configuration

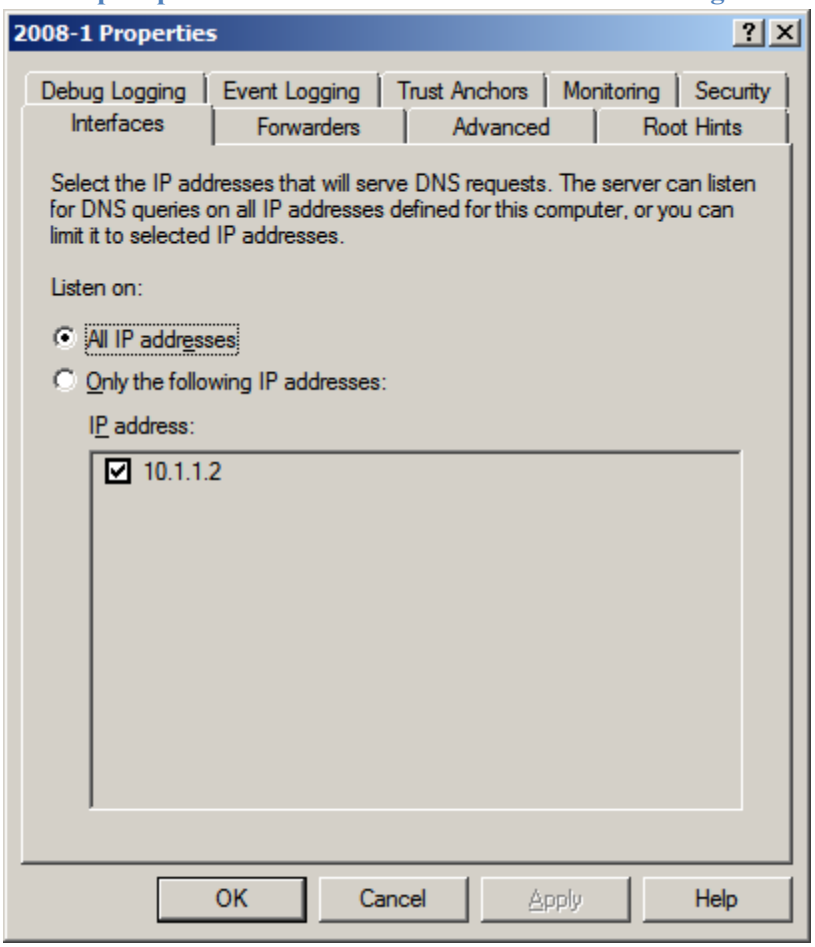


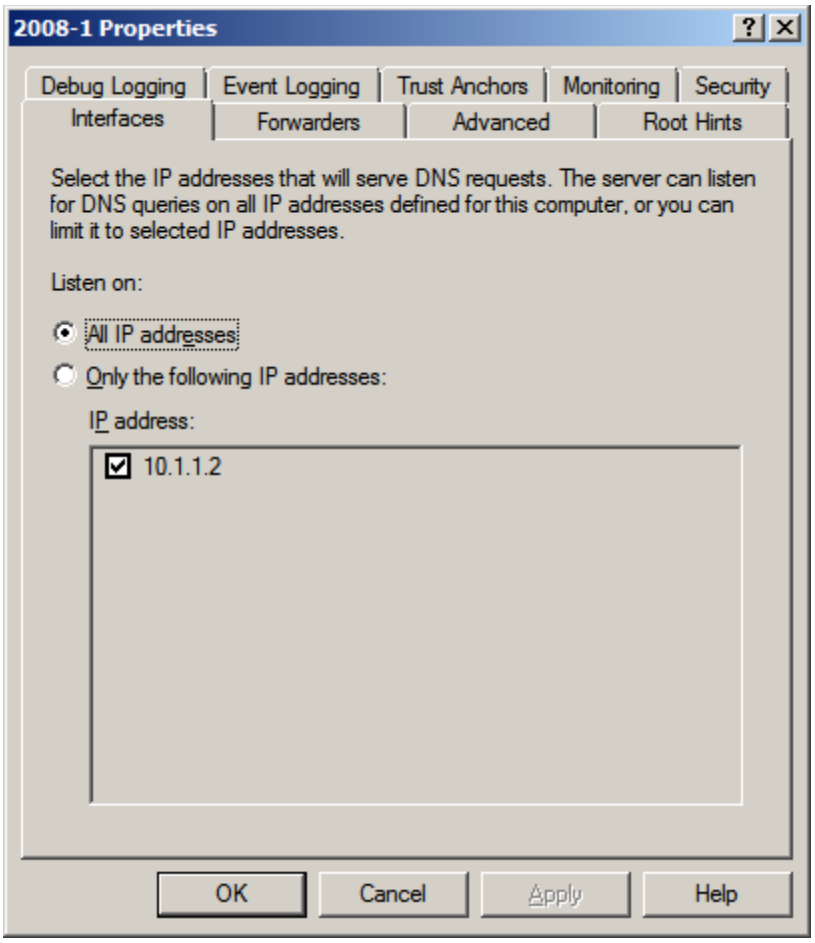
Appendix G – DNSv6 Issues

Appendix G – DHCP Scope Options – DNS Server



Appendix G – DHCP Scope Options – IPv6 Address of DNS Server Missing




2008-1 Properties
Debug Logging
Event Logging
Trust Anchors
Monitoring
Security
Interfaces
Forwarders
Advanced
Root Hints
Select the IP addresses that will serve DNS requests. The server can listen for DNS queries on all IP addresses defined for this computer, or you can limit it to selected IP addresses.
Listen on:
All IP addresses
Only the following IP addresses:
IP address:
10.1.1.2
OK
Cancel
Apply
Help

Appendix H – Switch Configuration with IPv4 and IPv6 Settings

Appendix H – Switch Configuration with IPv4 and IPv6 Settings

```
IPv4IPv6Lab#sh run
Building configuration...

Current configuration : 2342 bytes
!
! Last configuration change at 07:42:39 UTC Mon Mar 8 1993
!
version 15.0
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname IPv4IPv6Lab
!
boot-start-marker
boot-end-marker
!
!
enable password cisco
!
no aaa new-model
system mtu routing 1500
ip routing
!
!
!
ipv6 unicast-routing
!
!
!
!
!
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
!
!
!
!
!
```

```
!
!
!
!
!
!
!
!
!
!
interface FastEthernet0/1
 description server-1
 spanning-tree portfast
!
interface FastEthernet0/2
 description server-2
 switchport access vlan 2
 spanning-tree portfast
!
interface FastEthernet0/3
 description server-3
 switchport access vlan 3
 spanning-tree portfast
!
interface FastEthernet0/4
 description Access Point
 switchport access vlan 4
 spanning-tree portfast
!
interface FastEthernet0/5
 description laptop
 switchport access vlan 5
 spanning-tree portfast
!
interface FastEthernet0/6
 description Printer
 switchport access vlan 6
 spanning-tree portfast
!
interface FastEthernet0/7
 description laptop win7
 switchport access vlan 5
 spanning-tree portfast
!
interface FastEthernet0/8
 description SPAN port
```

```
 shutdown
!
interface GigabitEthernet0/1
 switchport access vlan 5
 spanning-tree portfast
!
interface Vlan1
 ip address 10.1.1.1 255.255.255.0
 ipv6 address 2001:DB8:85A3:1::1/64
!
interface Vlan2
 ip address 10.2.1.1 255.255.255.0
 ipv6 address 2001:DB8:85A3:2::1/64
!
interface Vlan3
 ip address 10.3.1.1 255.255.255.0
 ipv6 address 2001:DB8:85A3:3::1/64
!
interface Vlan4
 ip address 10.4.1.1 255.255.255.0
 ip helper-address 10.2.1.2
 ipv6 address 2001:DB8:85A3:4::1/64
 ipv6 nd managed-config-flag
 ipv6 nd other-config-flag
 ipv6 dhcp relay destination 2001:DB8:85A3:2::2
!
interface Vlan5
 ip address 10.5.1.1 255.255.255.0
 ip helper-address 10.2.1.2
 ipv6 address 2001:DB8:85A3:5::1/64
 ipv6 nd managed-config-flag
 ipv6 nd other-config-flag
 ipv6 dhcp relay destination 2001:DB8:85A3:2::2
!
interface Vlan6
 ip address 10.6.1.1 255.255.255.0
 ipv6 address 2001:DB8:85A3:6::1/64
!
ip http server
ip http secure-server
!
!
!
!
!
!
```

```
!
line con 0
line vty 0 4
 exec-timeout 0 0
 password cisco
 login
line vty 5 15
 password cisco
 login
!
!
monitor session 1 source vlan 1 - 6
monitor session 1 destination interface Fa0/8
end

IPv4IPv6Lab#
```

Appendix I – Switch Configuration IPv4 Removed

Appendix I – Switch Configuration IPv4 Removed

```
User Access Verification

Password:
IPv4IPv6Lab>en
Password:
IPv4IPv6Lab#sh run
Building configuration...

Current configuration : 2269 bytes
!
! Last configuration change at 07:47:31 UTC Mon Mar 8 1993
!
version 15.0
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname IPv4IPv6Lab
!
boot-start-marker
boot-end-marker
!
!
enable password cisco
!
no aaa new-model
system mtu routing 1500
!
!
!
ipv6 unicast-routing
!
!
!
!
!
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
!
```

```
!
!
!
!
!
!
!
!
!
!
!
!
!
interface FastEthernet0/1
 description server-1
 spanning-tree portfast
!
interface FastEthernet0/2
 description server-2
 switchport access vlan 2
 spanning-tree portfast
!
interface FastEthernet0/3
 description server-3
 switchport access vlan 3
 spanning-tree portfast
!
interface FastEthernet0/4
 description Access Point
 switchport access vlan 4
 spanning-tree portfast
!
interface FastEthernet0/5
 description laptop
 switchport access vlan 5
 spanning-tree portfast
!
interface FastEthernet0/6
 description Printer
 switchport access vlan 6
 spanning-tree portfast
!
interface FastEthernet0/7
 description laptop win7
 switchport access vlan 5
```

```
 spanning-tree portfast
!
interface FastEthernet0/8
 description SPAN port
 shutdown
!
interface GigabitEthernet0/1
 switchport access vlan 5
 spanning-tree portfast
!
interface Vlan1
 no ip address
 no ip route-cache
 ipv6 address 2001:DB8:85A3:1::1/64
!
interface Vlan2
 no ip address
 no ip route-cache
 ipv6 address 2001:DB8:85A3:2::1/64
!
interface Vlan3
 no ip address
 no ip route-cache
 ipv6 address 2001:DB8:85A3:3::1/64
!
interface Vlan4
 no ip address
 no ip route-cache
 ipv6 address 2001:DB8:85A3:4::1/64
 ipv6 nd managed-config-flag
 ipv6 nd other-config-flag
 ipv6 dhcp relay destination 2001:DB8:85A3:2::2
!
interface Vlan5
 no ip address
 no ip route-cache
 ipv6 address 2001:DB8:85A3:5::1/64
 ipv6 nd managed-config-flag
 ipv6 nd other-config-flag
 ipv6 dhcp relay destination 2001:DB8:85A3:2::2
!
interface Vlan6
 no ip address
 no ip route-cache
 ipv6 address 2001:DB8:85A3:6::1/64
!
```

```
ip http server
ip http secure-server
!
!
!
!
!
!
!
line con 0
line vty 0 4
 exec-timeout 0 0
 password cisco
 login
line vty 5 15
 password cisco
 login
!
!
monitor session 1 source vlan 1 - 6
monitor session 1 destination interface Fa0/8
end

IPv4IPv6Lab#
```

Appendix J – USE Case Diagram

Appendix J – USE Case Diagram